



Universitat Autònoma de Barcelona

ADOBE FORMS

Memòria del projecte
d'Enginyeria Tècnica en
Informàtica de Gestió
realitzat per
Laura Domènech Moreno
i dirigit per
Oscar Garcia Alcoceba

Escola Universitària d'Informàtica
Sabadell, juny de 2009

El/la sotasignant, **Daniel Franco**,
professor/a de l'Escola Universitària d'Informàtica de la UAB,

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva direcció
per en Laura Domènech Moreno
I per a que consti firma la present.
Sabadell, juny de 2009

Signat: Daniel Franco

El/la sotasignant, Oscar Garcia Alcoceba,
de “Everis”,

CERTIFICA:

Que el treball al que correspon la present memòria
ha estat realitzat sota la seva supervisió
per en Laura Domènech Moreno

I per a que consti firma la present.
Sabadell, juny de 2009

Signat: Oscar Garcia Alcoceba

Agraïments

Gràcies a Oscar Garcia
Alcoceba per donar-me l'oportunitat
de realitzar aquest projecte a
l'empresa "Everis", per la seva
paciència i pel seu suport en tot
moment en el moment del
desenvolupament d'aquest.

Resum contingut memòria

En aquesta memòria podem trobar l'explicació detallada del projecte final de carrera titulat "Adobe Forms".

Per començar, trobarem una introducció on es descriurà una visió general del projecte implementat. A continuació, en els objectius i antecedents, explicaré la situació en la que ens trobem actualment, els problemes existents i la situació a la que volem arribar, mitjançant el desenvolupament d'aquest projecte.

Una vegada el lector s'hagi fet una idea general, parlarem sobre la base teòrica que he necessitat per dur a terme la implementació, concretament, en aquest apartat, parlaré sobre la Gestió Documental; antecedents, el concepte de Gestió Documental, el desenvolupament d'aquesta, les fases per a la tractació de documents, els elements (bases de dades, hardware, xarxes, usuaris i administradors) i les avantatges de la implementació de la Gestió Documental. Posteriorment, parlaré sobre els documents intel·ligents, l'ús i avantatges d'aquests i la seva arquitectura, i seguidament, comentaré diferents aplicacions que gestionen la documentació documental, tals com, docShare, Neo-Doc, entre altres....

Una vegada el lector tingui una idea sobre la gestió dels documents i les facilitats que donen a les empreses la utilització de softwares que permetin tractar aquests, ens centrarem amb la plataforma Adobe LiveCycle, que és la que jo he utilitzat per crear el projecte final de carrera.

La plataforma LiveCycle esta formada per l'adobe LiveCycle Designer, l'adobe LiveCycle Reader Extensions, el Barcode i el Form Server. Al final d'aquest apartat, podrem observar una imatge per tal de que el lector es pugui fer una idea, de la manera amb que es relacionen tots aquests softwares.

A continuació, parlaré sobre temes que he necessitat per dissenyar la web. Aquest són servlets, jsp, Apache Tomcat, bases de dades i SQL.

Un cop el lector s'hagi fet una idea sobre la base teòrica utilitzada per el desenvolupament del projecte, entrarem a parlar sobre el disseny tècnic on el lector podrà veure els passos a seguir alhora del desenvolupament. Els passos més importants per al disseny són: el disseny del formulari, el disseny i configuració de la base de dades i per últim el disseny de l'aplicació web.

En el disseny tècnic, també parlarem sobre l'entorn tecnològic utilitzat, on podrem observar en una imatge l'arquitectura utilitzada, els llenguatges de programació, la descomposició en mòduls de les classes Java, i en la implementació descriurem el portal WEB dissenyat per realitzar una petita demostració sobre el procés general dels formularis.

Finalment, trobarem les conclusions generals, on explicaré els objectius assolits i també els coneixements adquirits, les línees de futur i la bibliografia.

Taula de continguts

Agraïments	4
Resum contingut memòria.....	5
Taula de continguts	7
1. Introducció.....	10
2. Objectius i Antecedents.....	11
3. Fonaments teòrics	12
3.1. Gestió Documental.....	12
3.1.1. Antecedents	12
3.1.2. Concepte.....	12
3.1.3. Desenvolupament de la gestió documental	13
3.1.4. Fases de la gestió documental.....	13
3.1.5. Elements Gestió Documental.....	14
3.1.6. Avantatges Gestió Documental	15
3.2. Documentació intel·ligent	17
3.2.1. Ús i avantatges dels documents intel·ligents.....	17
3.2.2. Arquitectura dels documents intel·ligents	18
3.2.3. Aplicacions disponibles en el mercat per a la Gestió Documental	19
docShare.....	19
Neo-Doc	19
Open-PDF.....	20
Microsoft Office Infopath.....	20
3.3. Plataforma Adobe LiveCycle.....	22
3.3.1. Adobe LiveCycle Designer.....	23
3.3.2. Adobe LiveCycle Barcode	24
3.3.3. Adobe LiveCycle Reader Extensions	26
3.3.4. Adobe LiveCycle Form Server	28
3.3.5. Cicle de vida Adobe LiveCycle	29
3.4. Servlets.....	30
3.4.1. Introducció.....	30
3.4.2. Cicle de vida d'un Servlet.....	31
3.4.3. ServletConfig i ServletContext.....	31

3.4.4.	Contenidors de Servlet	32
3.5.	JavaServer Pages.....	33
3.5.1.	JSP i Servlets.....	33
4.	Disseny tècnic	34
4.1.	Requeriments.....	35
4.1.1.	Requeriments funcionals generals.....	35
4.1.2.	Requeriments no funcionals	35
	Diagrama Casos d'Ús	36
4.2.	Fases projecte Adobe Forms	37
4.2.1.	Disseny dels formularis.....	37
4.2.2.	Disseny i configuració de la base de dades.....	43
4.2.3.	Disseny aplicació web	49
4.3.	Entorn tecnològic	51
4.3.1.	Entorn operatiu.....	51
4.3.2.	Entorn lògic de Programació	51
4.3.3.	Arquitectura	52
4.4.	Llenguatges utilitzats.....	54
4.4.1.	HTML	54
4.4.2.	CSS.....	54
4.4.3.	Javascript	54
4.4.4.	SQL.....	54
4.4.5.	Java	55
4.5.	Descomposició en mòduls	56
4.5.1.	Actions.....	56
4.5.2.	Interactions.....	59
4.5.3.	Beans.....	64
4.5.4.	Paràmetres	66
4.6.	Representació visual Projecte Adobe Forms	69
4.6.1.	Situació Actual.....	69
4.6.2.	Situació desitjada	70
4.7.	Descripció del portal WEB	71
4.8.	Portal WEB	72

4.8.1.	Accés Administrador	72
4.8.2.	Accés Usuaris	77
5.	Conclusions.....	82
6.	Futures línees d'actuació.....	84
7.	Bibliografia.....	85
7.1.	Fonts electròniques	85
7.1.1.	Gestio documental	85
7.1.2.	Documents intel·ligents	85
7.1.3.	Adobe.....	86
7.1.4.	Software gestió documental.....	87
7.1.5.	Tutorials Adobe.....	88
7.1.6.	Altres.....	89

1. Introducció

La tramitació de les Ajudes i Subvencions es considera un dels procediments claus sobre els quals s'ha d'implementar iniciatives d'Administració Electrònica i d'optimització de la gestió.

Actualment existeix una nova necessitat respecte als processos existents en la tramitació de les transaccions entre la Generalitat i els ciutadans, empreses i/o empleats públics; i és que, aquests tràmits no estiguin emmarcats en àmbit presencial; sinó que també es puguin suportar mitjançant vies telemàtiques.

Aquestes transaccions emmarcades en l'àmbit digital, requereixen l'ús de formularis electrònics que han de ser emplenats pels ciutadans, empreses privades, empleats públics a través d'una connexió segura a Internet.

Per tant, es pretén automatitzar les transaccions que es realitzen de forma manual amb l'objectiu de proporcionar serveis més senzills i eficients, agilitar els processos per a millorar el temps de resposta i reduir els costos, reduir la quantitat de paper que s'utilitza, facilitar la interoperabilitat entre els diferents organismes col·laboradors i assegurar l'intercanvi fiable i segur de dades i documents.

Aquest projecte té com a objectiu l'activació de formularis electrònics PDF, i es basa en les característiques pròpies de la Plataforma de Documents Intel·ligents d'Adobe, dins de la gamma LiveCycle.

2. Objectius i Antecedents

L'objectiu d'aquest projecte és desenvolupar un pilot amb la tecnologia ADOBE FORMS que implementi el servei de Formularis Electrònics.

Aquest projecte està incorporat dintre de la disciplina de la Gestió Documental i Tramitació de l'empresa "Everis" que sorgeix com a resposta a alts costos de gestió i emissió, ineficiències en la distribució i en l'emmagatzematge i posterior consulta dels documents.

Actualment es requereixen grans quantitats de personal humà o temps únicament per a la introducció manual de les dades dels documents en diversos sistemes (ERP, CRM,). Aquests realitzen tasques altament repetitives que no aporten valor a les organitzacions.

A part, la qualitat de les dades introduïdes en els diversos sistemes en general és baixa a causa del tractament manual de dades i no és possible a posteriori la visualització dels documents i l'accés als documents físics en paper és problemàtic. Per tant, amb aquest projecte pretenem automatitzar la captura de dades, l'ús de sistemes de millora dinàmica d'imatge i la comprovació automàtica de la coherència de les dades capturades.

Els avantatges d'aquesta solució serien l'elaboració del formulari per sol·licitar l'ajuda d'emancipació segons la plantilla de la Generalitat, el tractament de les dades recopilades (captura automàtica de dades del formulari rebut, sense cap acció addicional), la supressió de la necessitat del desplaçament físic dels sol·licitants; estalvi de temps i paper.

Amb la mateixa fiabilitat, aquesta solució també ens ha d'assegurar la protecció de dades amb signatura digital, certificacions i autenticacions, la possibilitat de disposar de forma unívoca del format paper a partir del digital, i del format digital a partir del paper (ús de codi de barres), la correcció de certs errors, ja que, el mateix programa comprova en temps real els camps emplenats, avisant en cas d'error.

Finalment, la solució també ens permetrà la facilitat d'integració amb altres aplicacions, ja que es treballa amb dades en format XML i la disponibilitat dels formularis les 24 hores del dia, 365 dies de l'any.

3. Fonaments teòrics

3.1. *Gestió Documental*

3.1.1. Antecedents

En alguns casos i en funció de la seva activitat, les organitzacions necessiten accedir i consultar de forma freqüent la informació arxivada. En uns altres, és el valor, la importància dels documents o, simplement, el volum d'informació el que empeny a buscar noves solucions innovadores que ofereixin avantatges i valor afegit sobre els sistemes tradicionals d'arxiu i emmagatzematge (fitxers, microfilmació, etc.).

Històricament han existit barreres d'accés per a dissenyar solucions eficients en aquest camp: escassa capacitat de la tecnologia documental, limitacions en les línies de comunicacions, manca d'eines específiques, falta d'integració... són factors que, units a l'alt cost, han impedit avançar i produir solucions competitives. No obstant això, en els últims anys hem assistit a un procés accelerat d'integració entre diverses tecnologies (dades, imatges, sons, etc.), de desenvolupament d'infraestructures de comunicació i d'abaratiment general de la tecnologia.

D'aquesta manera, sorgeixen les solucions i serveis de *Gestió Documental Digitalitzada* fruit de la fusió de productes físics i lògics de documentació, procés de dades, imatge i so, en un entorn de racionalització de recursos i mètodes de treball.

3.1.2. Concepte

La gestió documental és el conjunt de normes, tècniques i pràctiques utilitzades per a administrar el flux de documents de tot tipus en una organització, permetre la recuperació d'informació, determinar el temps en que els documents han de guardar-se, eliminar els que ja no serveixen i assegurar la conservació indefinida dels documents més valuosos, aplicant principis de racionalització i economia.

És una activitat gairebé tan antiga com l'escriptura, que va néixer a causa de la necessitat de documentar actes administratius i transaccions legals i comercials per escrit per a donar fe dels fets. Aquest tipus de documents es van plasmar successivament en taules d'argila, fulles de paper, pergamins i paper, la gestió del qual, es va anar fent més complexa a mesura que creixia la grandària de les fons documentals.

3.1.3. Desenvolupament de la gestió documental

Durant segles, la gestió documental en les organitzacions va ser el domini exclusiu d'administradors, arxivers i bibliotecaris, les eines bàsiques dels quals eren els llibres de registre, les carpetes, arxivadors, caixes i prestatgeries on es guarden els documents de paper (i més tard els audiovisuals i els documents en suports magnètics o òptics), els fitxers que permeten fer referències creuades i una llarga llista de tècniques de recuperació d'informació mitjançant sistemes de codificació i classificació.

Recentment, es van afegir els documents informàtics, que són cada vegada més necessaris a causa de la complexitat i nivell de sofisticació que van arribant als sistemes computacionals de suport de l'activitat administrativa.

L'ús del computador en la gestió documental s'inicia a partir de les grans biblioteques nacionals anglòfones, la Biblioteca del Congrés dels Estats Units d'Amèrica i la "British Library", que en els anys 60 del segle XX creen programes de bases de dades coneguts com MARC (Machine Readable Cataloguing).

Quan l'ús de les tecnologies d'informació i comunicació es va fer comuna en l'administració pública i privada, amb el naixement de les bases de dades i l'aparició dels processadors de textos i altres aplicacions ofimàtiques, i sobretot amb l'arribada del correu electrònic, va sorgir la necessitat de capturar i conservar també documents que neixen, viuen i moren en format electrònic. Aconseguir això, va representar un nou salt en la complexitat i exigències als sistemes informatitzats i en la forma de pensar dels administradors.

En l'actualitat, coexisteixen en el món diversos sistemes de gestió documental: des del simple registre manual de la correspondència que entra i surt, fins als més sofisticats sistemes informàtics que manipulen no només la documentació administrativa pròpiament, sinó que a més controlen els fluxos de treball del procés de tramitació dels expedients, capturen informació des de bases de dades de producció, comptabilitat i altres, enllacen amb el contingut d'arxius, biblioteques, centres de documentació i permeten realitzar recerques sofisticades i recuperar informació de qualsevol lloc.

3.1.4. Fases de la gestió documental

Elaboració/creació documents.

Captura de la informació / documentació.

Emmagatzematge.

Anàlisi i tractament de la informació.

Gestió dels documents.

Publicació de la informació.

3.1.5. Elements Gestió Documental

3.1.5.1. Bases de dades

L'aplicació de la informàtica en tots els nivells de les organitzacions comprèn un fenomen que té les seves implicacions en la gestió documental: la proliferació de bases de dades sobre diferents aspectes, que en alguns casos estan substituint als documents com suport d'informació valuosa per a l'organització. Aquesta tendència s'aguditzava encara més en un entorn d'"i-business" en el qual les relacions amb els clients es realitzen a través d'Internet.

Des d'un punt de vista de gestió documental, un dels principals problemes és com identificar els documents dintre de l'entorn de la BD, saber quina part del contingut constituïx els documents que s'han de gestionar. Referent a això existeixen diverses opinions. D'una banda, hi ha qui pensa que una BD simplement conté informació per a realitzar càlculs o operacions. En l'altre extrem, està l'opinió que una BD és el resultat de les activitats de l'empresa i és per això pel que les hi pot considerar documents de gestió.

3.1.5.2. Hardware

Escàners i dispositius de Digitalització: Els documents originals, una vegada preparats, són transformats en documents digitals, els quals seran guardats o emmagatzemats.

Servidors: Contenen la informació prèviament digitalitzada. Els usuaris finals es connectaran a un servidor per a poder accedir a aquesta informació, per a la seva modificació o revisió.

3.1.5.3. Xarxes

Per mitjà de les xarxes els usuaris podran accedir a la informació que es troba en els servidors. Les xarxes poden ser locals, encara que, també es pot accedir a la informació per Internet.

3.1.5.4. Usuaris

A través d'un compte d'Usuari, es porta a terme l'accés als documents digitalitzats dintre del Sistema de Gestió Documental permetent així, realitzar la consulta electrònica dels mateixos d'acord als nivells de seguretat assignats a cadascun dels usuaris registrats en el Sistema.

3.1.4.5. Administradors

Des del lloc de l'Administrador del Sistema, els documents digitalitzats es codifiquen i indexen dins la base de dades del servidor, identificant la ubicació física del document original i assignant, a cada document lògic, les claus d'accés. Els documents digitalitzats s'emmagatzemen en el seu corresponent fitxer.

3.1.6. Avantatges Gestió Documental

3.1.6.1. Control efectiu: senzillesa, rapidesa i estalvi

D'una forma senzilla, l'organització té accés instantani a tota la documentació necessària per a la seva activitat de negoci, amb els avantatges afegits de l'eliminació de desplaçaments, reducció de temps de consultes i tasques d'arxiu, estalvi d'espai físic, resolució del problema de localització de documents...

3.1.6.2. Ús racional dels recursos

La gestió documental facilita que la informació es comparteixi i s'aprofiti de forma més eficient i com un recurs col·lectiu. Com a conseqüència, es redueixen dràsticament situacions com la duplictat de documents arxivats, fotocòpies innecessàries, dobles enregistraments de dades, etc.

3.1.6.3. Seguretat i fiabilitat.

Informació, documents, etc. de gran valor per a l'organització poden custodiar-ne en locals d'alta seguretat, garantint el seu perfecte estat de conservació.

3.1.6.4. Productivitat i valor afegit

Una gestió documental, a part d'estalvi de costos, genera una productivitat i valor afegit addicionals, originats pel ràpid accés a la informació dintre de l'organització i la seva posterior distribució, sense necessitat de traslladar els documents.

3.2. Documentació intel·ligent

La documentació intel·ligent són solucions desenvolupades que associen la lògica empresarial i les accions personalitzades amb el contingut d'un document. Els documents intel·ligents poden ser molt més que reposadors d'informació proporcionada per l'usuari, poden ser programats perquè siguin intel·ligents. Aquests poden introduir dades de forma automàtica, facilitant la tasca d'emplenar formularis o de treballar amb plantilles als treballadors de la informació. Els documents intel·ligents també poden tenir accés a dades externes de forma automàtica i col·locar-los correctament en un document, o bé proporcionar ajuda contextual per a guiar als treballadors de la informació en la preparació de documents complexos. La informació d'un document intel·ligent es pot compartir fins i tot en sistemes i aplicacions que no es troben en el mateix PC.

3.2.1. Ús i avantatges dels documents intel·ligents

Els documents intel·ligents assignen accions o controls a seccions concretes d'un document (delimitades amb etiquetes XML). Dintre d'aquestes seccions, el document intel·ligent, a través del panell de tasques, pot presentar informació als treballadors de la informació en forma de gràfics, de text d'ajuda, de fragments de documents o de controls basats en accions com quadres de text, botons d'opció, botons de comandaments o fins i tot, controls personalitzats.

Existeixen diversos avantatges en una solució de documents intel·ligents, algunes de les quals es descriuen a continuació:

Permet el flux de dades dintre d'una organització: a partir d'ara, la informació dels documents ja no seran dades mortes, que no es poden consultar ni processar de forma externa: les solucions de documents intel·ligents ajuden a integrar les dades dels documents en el magatzem de dades de l'empresa.

Els desenvolupadors poden definir la manera en la qual aquestes dades estan disponibles, incorporant la lògica empresarial i el model de seguretat de l'organització al funcionament del document intel·ligent.

Ajuda en la creació de documents estructurats: les solucions de documents intel·ligents són més adequades per a documents estructurats, com documents creats a partir de plantilles, revisions d'empleats, a més de qualsevol tipus de documentació com documents de requisits, documentació d'ajuda, notes de productes, etc.

Aquests es poden dissenyar perquè sigui auto complert gran part del

document en funció de les dades introduïdes per l'usuari o del comportament del mateix. S'encarrega de la seva pròpia distribució i de les actualitzacions: un mètode de distribució centralitzat i eficaç permet que l'administrador no hagi de visitar cadascun dels escriptoris per a garantir que la instal·lació és correcta. Una vegada que s'obre el document inicial o la plantilla en un equip client, el document intel·ligent es pot implementar i, posteriorment, actualitzar directament des del servidor sense intervenció alguna per part de l'administrador. Això sí, com el document intel·ligent consta de codi signat, els administradors tenen el control de la distribució de les actualitzacions.

3.2.2. Arquitectura dels documents intel·ligents

Els documents intel·ligents permeten als desenvolupadors assignar accions concretes a elements XML dins d'un document ja existent o d'un nou. El codi XML dintre del document assignat es correspon amb un esquema XML subjacent. Una vegada que el document està preparat, els desenvolupadors poden utilitzar la interfície API de documents intel·ligents, disponible com part del "kit" per a desenvolupadors per a assignar les accions que controlaran la solució.

Els desenvolupadors tenen molta flexibilitat a l'hora de treballar amb documents intel·ligents. Poden manipular el document directament o interactuar amb processos del servidor, com recuperar dades o guardar-los en una altre ubicació perquè els utilitzi un servidor. Els desenvolupadors també disposen de molta flexibilitat quant a la manera de desenvolupar: poden utilitzar Visual Basic 6.0, Visual Basic .NET, C# o C++. Una vegada compilat l'arxiu DLL d'un document intel·ligent, el desenvolupador ha de crear un arxiu de manifest que descriu la ubicació dels següents elements:

- Un arxiu DLL que contingui el codi d'automatització que controla la solució.
- Un esquema XML que es correspongui amb els elements assignats dintre del propi document.
- El Manifest XML que ajuda a protegir qualsevol element complex dels treballadors de la informació.

3.2.3. Aplicacions disponibles en el mercat per a la Gestió Documental

docShare

És una aplicació destinada a gestionar el flux de la informació dins de la seva empresa. Permet controlar els documents des de la creació, fins a l'emmagatzematge final, passant per les diferents modificacions per part dels usuaris que aquest document hagi patit al llarg del seu cicle de vida. Una de les característiques principals del gestor, és la utilització del contingut dels documents per a fins de recerca i de seguretat.

Neo-Doc

Amb neoDOC (la Gestió Documental de neoCS) és possible compartir documents i informació en temps real entre diverses seus o fins i tot amb clients, garantint la seguretat i la privadesa, i utilitzant sempre un entorn web, el que permet treballar via intranet/extranet des de qualsevol equip, sense necessitat de cap instal·lació client i amb qualsevol sistema operatiu o arquitectura.

La finalitat és satisfer la contínua demanda d'informació que es requereix en els diferents contextos dels objectius de cada empresa.

La Gestió Documental neoDOC fomenta una organització basada en una eficient col·laboració, coordinació i comunicació, on la informació flueix més ràpid i amb precisió, eliminant moltes barreres entre departaments, augmentant la productivitat dels treballadors, millorant els processos comercials i el servei al client, permetent una ràpida presa de decisions i agilitant el procés de negoci global.

Això s'aconsegueix amb una reducció del temps emprat en els processos diaris de captura i indexació de documents, recuperació i seguiment d'informació, gestió de continguts, lliurament de notificacions i automatització de fluxos de treball en assignació de tasques, supervisió del procés i registres. En altres paraules, reducció de costos i augment de l'eficiència.

Open-PDF

Es basa en la gestió de formularis intel·ligents en PDF.

El format PDF s'ha convertit en un estàndard de visualització, intercanvi i impressió de documents.

Actualment la funcionalitat d'aquest format s'ha ampliat per suportar la incorporació de formularis que permeten introduir dades en el mateix document. OpenTrends ha desenvolupat un conjunt de mòduls i aplicacions de software lliure que permeten a les empreses:

- Crear documents i formularis de forma ràpida, incorporant validacions de formats, camps variables i validacions en Javascript.
- Visualitzar els formularis i modificar-los.
- Incorporar anotacions adhesives, icones, notes manuals.
- Entrada de dades en els formularis amb possibilitat de gravació del formulari en disc local.
- Signatura digital.
- Incorporació de fitxers annexos.
- Generació de codis de barres en diversos formats incloent-hi codis 2D (PDF417).
- Extracció dels dades del fitxer PDF i validació de la signatura digital.
- Escanejat i reconeixement de codis de barres.

Aquesta tecnologia, openPDF, està basada en diversos productes de programari lliure, Scribus, multivalent, PDFBox, iText i PapierMaché entre altres.

Microsoft Office Infopath

Microsoft Office InfoPath 2007 és un programa de recopilació d'informació inclòs en la versió 2007 Microsoft Office Systems. Office InfoPath 2007 permet crear i implementar solucions de formularis electrònics per a recopilar informació d'una manera eficient. També pot utilitzar les capacitats de InfoPath Forms Services en Microsoft Office SharePoint Server 2007 per a estendre els processos empresarials més enllà del servidor de seguretat de l'empresa, enviant els formularis en forma de missatges de correu electrònic de Microsoft Office Outlook, formularis d'explorador Web o formularis per a dispositius mòbils.

Office InfoPath 2007 millora l'eficàcia de la recopilació d'informació enviant formularis electrònics que arriben més fàcilment als participants , redueixen l'entrada de dades repetides i milloren la qualitat de les dades obtingudes. Les solucions de formularis de InfoPath permeten arribar a més usuaris facilitant el procés d'emplenar els formularis en exploradors Web, missatges de correu electrònic o dispositius mòbils.

Els formularis de InfoPath es poden distribuir en forma de missatges de correu electrònic per a accelerar el procés alhora d'omplir el formulari.

Office InfoPath 2007 ajuda a obtenir la informació correcta al primer intent gràcies l'ocupació de validació en temps real, camps emplenats prèviament i connexions a fonts d'informació importants. Les solucions de Office InfoPath 2007 estan totalment integrades en Microsoft Office Systems per a proporcionar capacitats eficaces de formularis electrònics en un paquet familiar.

Els formularis de InfoPath es poden implementar en diferents exploradors, entre uns altres, Microsoft Internet Explorer, Netscape, Mozilla i Safari.

InfoPath Forms Services de Office SharePoint Server 2007 facilita el control de les solucions de formularis proporcionant una administració centralitzada dels formularis electrònics per a tota l'organització. Office SharePoint Server 2007 proporciona un portal unificat i una solució de formularis i administració de contingut per a l'empresa que facilita l'administració de formularis electrònics en el mateix context que altres funcions empresarials essencials.

InfoPath Forms Services contribueix a minimitzar el temps d'inactivitat de la solució proporcionant actualitzacions de fàcil instal·lació, administració de versions i eines de supervisió. També permet la simplificació dels processos empresarials basats en formularis Office InfoPath 2007 i s'integra fàcilment en els sistemes i aplicacions empresarials actuals gràcies a l'ús d'estàndards industrials que permeten automatitzar els processos empresarials existents sense haver de reintentar-los.

3.3. *Plataforma Adobe LiveCycle*

Plataforma LiveCycle Adobe LiveCycle Enterprise Suite és una família integrada de programes per a automatitzar processos que ajuden als negocis i administracions públiques a relacionar-se amb major eficàcia amb els clients, ciutadans, socis i proveïdors.

Amb LiveCycle, les organitzacions poden distribuir aplicacions orientades a l'exterior amb les quals és més fàcil interactuar. Això permet a les companyies comunicar-se millor amb aquelles persones que poden sentir-se frustrades o confoses amb els procediments en línia, i que probablement abandonaran les transaccions, recorrent a recursos de major cost tals com visites personals o assistència telefònica. Al transformar processos com obertura de comptes, processat de reclamacions o auto ajuda guiada en aplicacions atractives, les empreses i administracions públiques poden millorar el servei al client, reduir la costosa durada dels cicles, i gestionar informació de forma més ràpida i precisa. La plataforma ADOBE LiveCycle permet gestionar i servir formularis PDF I HTML, així com gestionar les interaccions i enviaments amb el servidor.

Els serveis inclosos de la plataforma de formularis ADOBE són:

Generació de documents:

Permet generar de forma dinàmica documents d'alta qualitat utilitzant dades procedents de les diverses aplicacions de negoci i diferents formats de fitxer. Aquesta generació pot ser tipus "batch" (processament per lots) o en temps real.

Col·laboració:

Permet que diverses persones treballin junts creant, revisant i marcant documents i intercanviant informació de negoci.

Gestió de processos:

Permet automatitzar els fluxos de treball basats en documents, accelerant processos de negoci crítics.

Gestió i control de la seguretat dels documents:

Gestionar tant l'accés a la informació crítica de negoci, així com certificant l'autenticitat del document per mitjà de signatura digital.

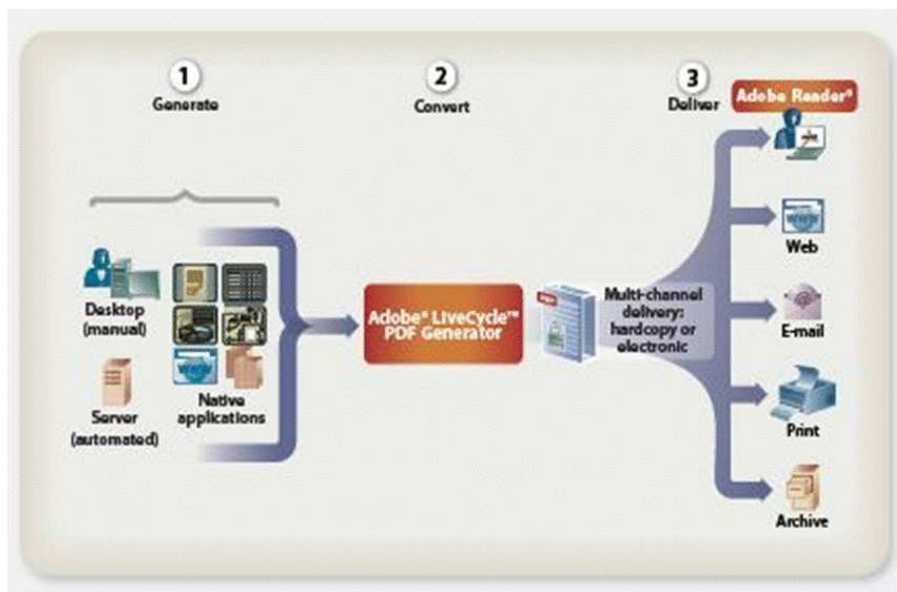
Dins del cicle de vida d'un formulari intervenen diferents components de l'arquitectura ADOBE.

3.3.1. Adobe LiveCycle Designer

Disseny de plantilles de formularis a ser utilitzats per altres solucions. És una eina fàcil d'utilitzar que permet dissenyar còpies exactes de documents en paper incloent logotips, línies, requadres, ombres, codi de barres, camps de data variable...

Independitza a les aplicacions de la gràfica dels documents, la qual, pot ser modificada amb aquesta eina sense la necessitat de canvi en la programació. El disseny de plantilles i el seu emmagatzematge en arxius magnètics permet assolir estalvis en els processos administratius d'adquisicions i emmagatzematge físic de papereria.

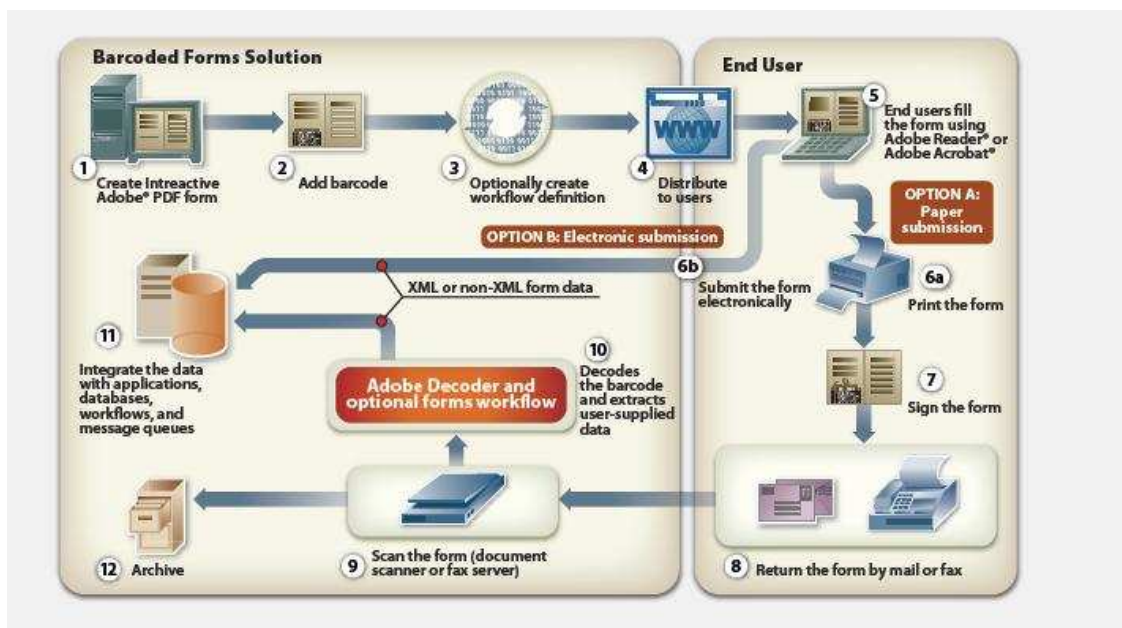
La creació de formularis electrònics s'estén des de formularis estàtics a dinàmics. Sobre la base de plantilles en altres formats diferents de PDF, es pot transformar a documents PDF (Word Excel, Power Point, entre altres). L'estructura de dades incorporades a l'interior del formulari poden dur la formulació de regles de negoci. El disseny dels formularis té les eines necessàries per a incorporar camps específics per a ser utilitzats com contenidors de signatura digital, encriptar dades i/o el formulari, validar signatures.



Procés AdobeLiveCycle Designer

3.3.2. Adobe LiveCycle Barcode

Permet afegir en un formulari PDF, un codi de barres bidimensional PDF417 que conté les dades que l'usuari hagi introduït en el formulari i d'aquesta forma poder bolcar de forma automàtica al sistema o Base de dades. Aquest programa permet tenir la possibilitat de tenir en convivència les maneres de tramitació telemàtica i presencial. Els avantatges d'aquest programa són la possibilitat d'automatitzar la captura d'informació del formulari, automatitzar també el tractament de la informació rebuda sense la necessitat de cap intervenció addicional, facilita la transacció progressiva del paper a la sol·licitud electrònica i elimina la necessitat que la informació s'hagi de digitalitzar una segona vegada per a tenir-la a la disposició de les altres aplicacions.



Procés que segueix un document digital amb codi de barres

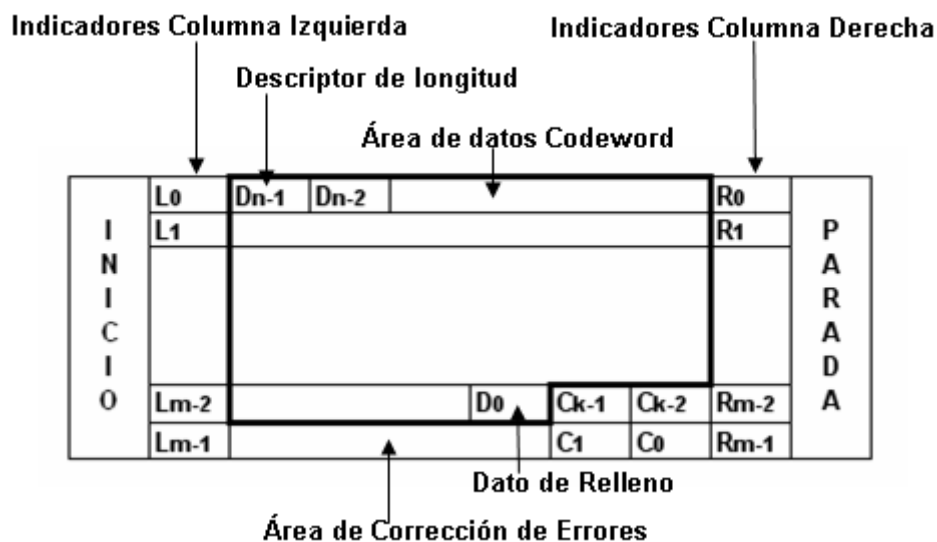
PDF417

El PDF417 més que un Codi de Barres, és un fitxer portàtil de dades. Tradicionalment, els codis de barres, han contingut una petita quantitat d'informació, funcionant com si fossin "claus" per a l'accés a fitxers emmagatzemats en el computador principal. Per si mateixos, no contenen usualment informació significativa: sense la base de dades externa, els codis de barres no tenen significat real.

Però el PDF417, ha desplaçat aquesta barrera de codi de barres, per a poder treballar com un fitxer portàtil de dades. Amb aquest mitjà d'alta capacitat, es pot accedir immediatament a les dades virtualment en qualsevol aplicació.

És un mètode de transport de dades capturades per escàner; elimina la pèrdua de temps i la introducció de dades de forma manual i amb possibilitat d'errors. Funciona com una memòria de computador, que pren com base un paper d'un sol ús que pot escriure's sobre ell mateix i pot llegir-se moltes vegades dintre d'una àmplia varietat d'aplicacions. El mateix que un llenguatge màquina, el PDF417, estableix comunicació amb tots els sistemes operatius del sistema, sense importar la classe de computador que es disposi. A més pot imprimir-se amb qualsevol de les tecnologies actuals, fins i tot amb impressores portàtils.

És un codi multi files, continu, de longitud variable, que té alta capacitat d'emmagatzematge de dades. El codi consisteix en un patró de marques (17,4), els apartats estan definits en termes de valors particulars d'una funció discriminatòria, cada apartat inclou 929 codewords (925 per a dades, 1 per als descriptors de longitud i almenys 2 per a la correcció d'error) disponibles i té un mètode de dos passos per a descodificar les dades escanejades. L'ample de la barra més prima és cridada mòdul, després d'això un mòdul de la barra se simbolitza amb un (1) i un mòdul d'espai amb un zero (0). El codi aquesta compost entre 3 a 90 files. Una fila aquesta composta entre 1 i 30 columnes de dades i la seva ample varia entre 90 i 583 mòduls amb les marges. Els nivells de correcció d'errors varia entre zero (0) i vuit (8), la correcció comprèn entre 2 (nivell 0) i 512 (nivell 8) codewords (CW).



Descripció contingut Codi de Barres PDF417



Codi de Barres PDF417

3.3.3. Adobe LiveCycle Reader Extensions

Programa que permet afegir drets als documents Adobe PDF , activant funcionalitats en Adobe Reader, pel que , els usuaris del document podran gravar-lo, completar-lo, adjuntar altres arxius, signar-lo, agregar comentaris, enviar formularis PDF interactius, importar/exportar dades XML.

Aquestes funcionalitats s'activaran automàticament quan el destinatari obri el document i es desactivaran al tancar-lo.

Facilita la col·laboració i revisió de documents, ja que, usuaris interns i externs poden visualitzar, completar, agregar comentaris i arxius adjunts i retornar documents i formularis electrònic.

Disminueix costos, a l'eliminar processos basats en paper, redueix el temps i els costos associats a la impressió, distribució, procés i arxiu de paper.

La captura de dades és segura, ja que, els usuaris poden signar, certificar i autenticar els documents abans de retornar-los, assegurant la seva confidencialitat. Els usuaris també poden completar i signar un formulari online o offline guardant una còpia local per al seu propi registre.

Una captura de dades més fàcil i precisa redunda que els usuaris rebin un millor servei. Aquest programa també protegeix la seguretat i privadesa d'informació confidencial i millora la seguretat de les comunicacions a través de signatura, certificació i autenticació de documents i formularis utilitzant estàndards del mercat tals com Verisign, Entrust, GeoTrust...

Adobe LiveCycle Reader Extensions permet capturar dades crítiques del procés de negoci en formularis PDF intel·ligents, millorant l'exactitud de la informació a través de captura intel·ligent. Després, és possible incorporar aquestes dades directament als sistemes a través de transferència d'arxius XML.



figura 1.- Arquitectura A

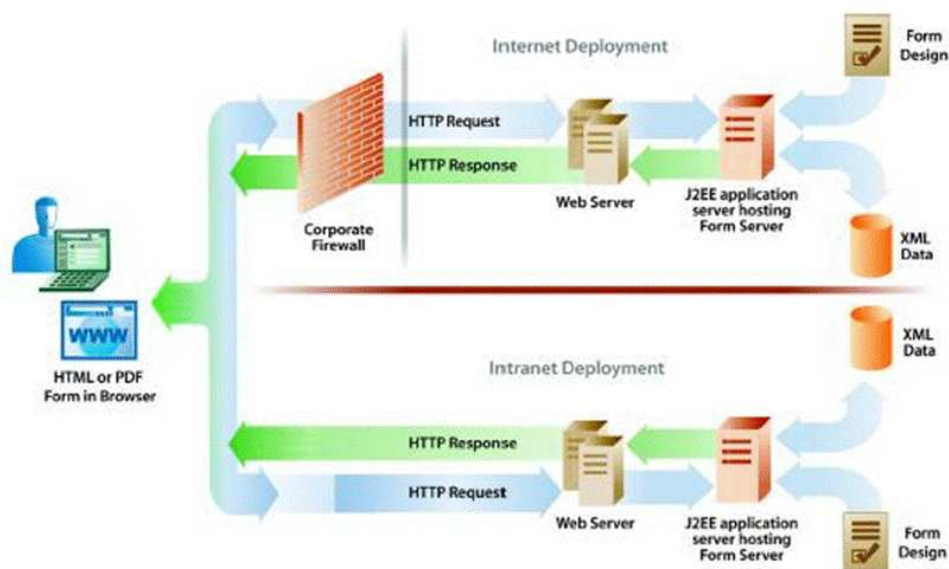
Arquitectura Adobe LiveCycle Reader Extensions

3.3.4. Adobe LiveCycle Form Server

Programa que estén processos basats en formularis més enllà de les fronteres de l'organització creant i distribuint formularis PDF o HTML des de plantilles XML dissenyats amb Adobe Designer.

Quan un usuari sol·licita un formulari, Form Server detecta la sol·licitud i el tipus de plataforma de l'usuari final i, basat en aquesta informació, transforma en forma dinàmica la plantilla en XML en un format HTML que millor s'adapti a l'entorn de l'usuari. Amb Adobe Form Server, les organitzacions poden connectar els sistemes empresarials a usuaris dintre i fora del Firewall, agilitant així la captura de dades de forma segura.

Amb Form Server, els formularis i la informació dels sistemes empresarials poden ser estesos a clients, proveïdors, empleats i socis, independent de la plataforma i dels dispositius que tinguin. S'integra fàcilment a les aplicacions web de l'organització proveint una interfície SOAP (Service-Oriented Architecture Protocol) i una interfície de programació d'aplicacions.

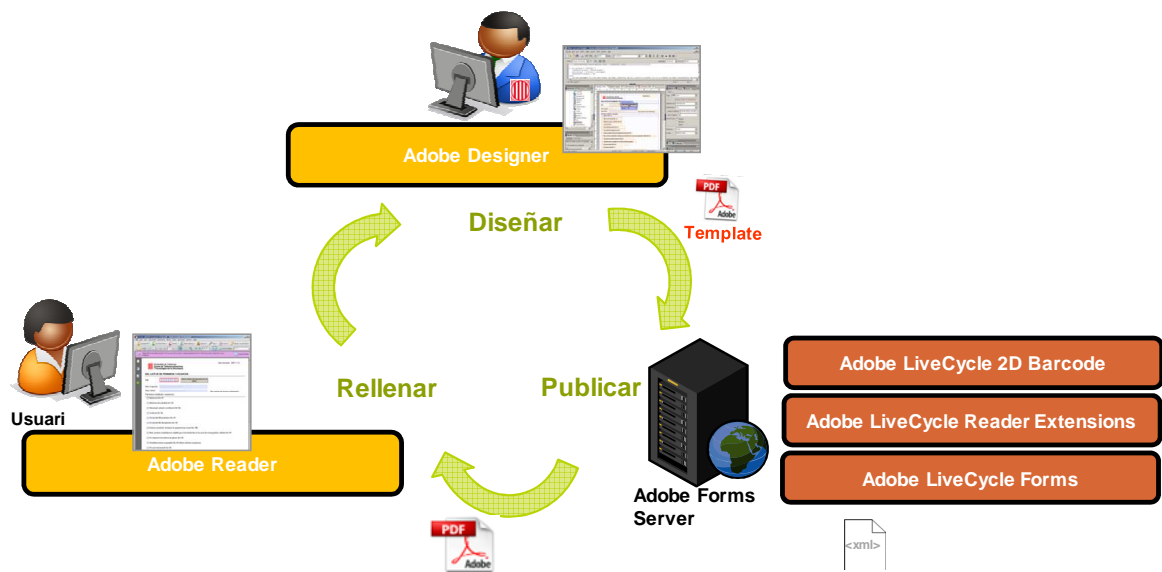


Arquitectura Adobe LiveCycle Form Server

3.3.5. Cicle de vida Adobe LiveCycle

Dins del cicle de vida d'un formulari intervenen les diferents parts anteriorment descrites.

En la següent figura es mostren de forma esquemàtica aquestes relacions.



Cicle de vida AdobeLiveCycle

3.4. Servlets

3.4.1. Introducció

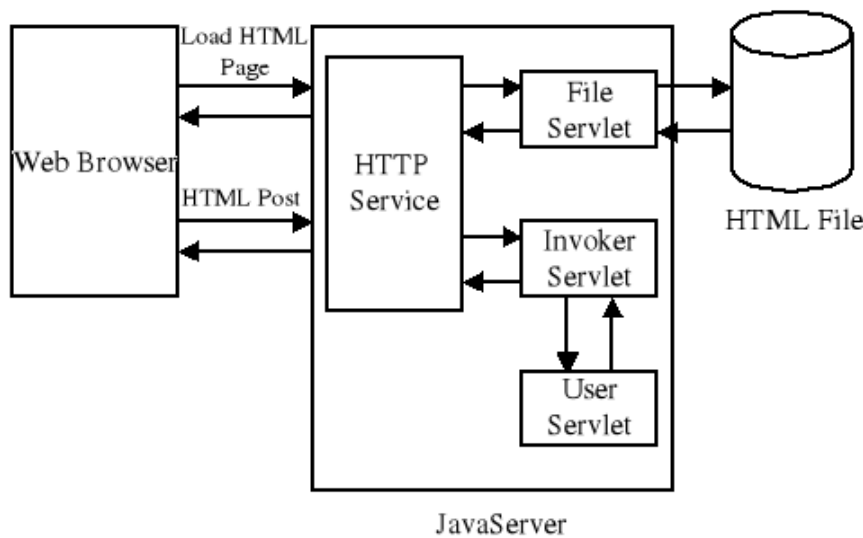
Les miniaplicacions de servidor (anglès servlets) són objectes Java executats per un servidor d'aplicacions i que responen a invocacions HTTP, servint pàgines dinàmiques.

El contingut generat pot ser un fitxer de qualsevol tipus, la majoria de vegades HTML.

Un objecte servlet és capaç de rebre una invocació i generar una resposta en funció de les dades de la invocació, de l'estat del propi sistema i les dades a què pugui accedir. El paquet bàsic de servlet defineix objectes Java que maneguen peticions i respostes fetes per servlets, així com objectes que informen dels paràmetres de configuració del servlet i l'entorn d'execució. El paquet `javax.servlet.http` defineix subclasses específiques HTTP dels elements genèrics del servlet, com ara objectes de gestió de sessions que lliguen múltiples peticions del mateix client i respostes del servidor d'aplicacions. Els servlets poden estar empaquetats dintre d'un fitxer de format WAR, com a aplicació Web, dintre d'un contenidor.

L'execució d'un servlet es fa dintre d'un o més processos del servidor d'aplicacions, de manera que es genera un nou flux. No generar un nou procés (com acostumen a fer els CGIs) implica un estalvi de recursos que es tradueix en un millor rendiment del sistema.

Els servlets poden ser objectes java precompilats o JSP compilats en temps d'execució (o en un altre moment després de l'arrencada del servidor d'aplicacions).



Funcionament Servlets

3.4.2. Cicle de vida d'un Servlet

Consisteix en les passes següents:

La instància de Servlet és carregada en temps d'arrencada del contenidor.

El contenidor crida el mètode `init()`, en temps d'arrencada, o quan rep la notificació de la primera invocació. Aquí és on s'inicialitza el servlet i ha de ser executat abans que se serveixi cap petició. Aquest mètode no torna a ser executat en tota l'execució del contenidor.

Després de la inicialització el servlet pot començar a servir invocacions de clients, cadascuna servida pel seu propi fil d'execució (Thread). Per cada invocació s'executa el mètode `service()`.

Finalment, en temps d'aturada del contenidor, aquest invoca el mètode `destroy()`.

3.4.3. ServletConfig i ServletContext

En un contenidor hi ha una sola instància de `ServletContext`. Aquest objecte pot ser usat per qualsevol instància de Servlet informació a nivell de l'aplicació o detalls sobre el contenidor. Cada servlet, per altra banda, obté la seva pròpia instància de `ServletConfig`. Aquest objecte proveeix paràmetres d'inicialització per un servlet. Un desenvolupador pot obtenir la referència a `ServletContext` usant qualsevol d'aquests dos objectes.

3.4.4. Contenidors de Servlet

Un contenidor de Servlet és un servidor web especialitzat que suporta l'execució de classes Servlet. Combina la funcionalitat bàsica d'un servidor web amb optimitzacions i extensions específiques de Java i de Servlet, com la integració amb l'entorn d'execució Java i la capacitat de traduir específicament URLs en peticions de Servlet. Els servlets són registrats pels contenidors de servlets, que es nodreixen de la informació sobre les funcionalitats que aquests proveeixen i quina URL usaran per identificar-se ells mateixos. A partir d'aquest moment, el contenidor és capaç d'inicialitzar el Servlet i respondre a invocacions conforme van arribant. Molts contenidors tenen la capacitat d'afegir i treure servlets de forma dinàmica, avantatge que permet que afegir, modificar o treure servlets no afecti la resta de servlets. Els contenidors de Servlet també poden ser anomenats contenidors web o motors web.

Com en altres APIs de Java, diferents companyies proveïdores posen a disposició la seva pròpia implementació de contenidor Servlet estàndard. A sota hi ha una llista de contenidors web, tant lliures com comercials. En aquests casos 'lliure' es refereix a què poden ser usats de forma gratuïta fins i tot amb finalitats lucratives.

3.5. *JavaServer Pages*

JavaServer Pages és una tecnologia que permet als desenvolupadors de pàgines web, generar respostes dinàmicament a peticions HTTP. La tecnologia permet que codi Java i certes accions predefinides siguin incrustades en un context estàtic.

La sintaxi de JSP incorpora tags XML addicionals, anomenats accions de JSP, per ser usats per invocar altres funcions. Addicionalment, la tecnologia permet la creació de llibreries d'etiquetes que actuen com extensions de l'estàndard d'etiquetes HTML o XML. Les llibreries d'etiquetes aporten una forma multi plataforma d'ampliar les capacitats d'un servidor web.

Els JSPs són compilats en Servlets per un compilador JSP. Aquest pot generar un servlet o generar bytecode directament.

3.5.1. JSP i Servlets

Des del punt de vista de l'arquitectura, els JSP poden ser vistos com una abstracció, d'alt nivell, de servlets implementada com una extensió de l'API Servlet 2.1. Ambdues tecnologies van ser inicialment desenvolupades per Sun Microsystems. Començant per la versió 1.2 de l'especificació de JSP, JavaServer Pages van ser desenvolupats sota la Java Community Process.

4. Disseny tècnic

La tramitació de les Ajudes i Subvencions es considera un dels procediments claus on implementar iniciatives d'Administració Electrònica i d'optimització de la Gestió.

Actualment, aquestes sol·licituds es fan de forma presencial, i per a qualsevol actuació es necessari personal que es dediqui al tractament de les dades.

Els formularis electrònics automatitzaran les transaccions que es fan de manera manual amb l'objectiu de proporcionar serveis més senzills i eficients, agilitzar els processos per millorar el temps de resposta i reduir els costos i la quantitat de paper que s'utilitza.

La sol·licitud per la tramitació telemàtica es dissenyarà utilitzant productes d'Adobe LiveCycle.

En el moment en que el ciutadà accedeixi al formulari de l'aplicació, aquesta generarà en aquest moment el document per al usuari de manera que totes les seves dades estaran actualitzades.

Per la generació d'aquests documents actualitzats, s'utilitzarà el producte Adobe LiveCycle Forms, encarregat de generar documents PDF. Per facilitar l'accessibilitat al document, els documents seran tractats pel producte Adobe LiveCycle Reader Extensions, mitjançant el qual, permetrà que el ciutadà pugui omplir el formulari, firmar i enviar el document, mitjançant l'Adobe Reader.

4.1. *Requeriments*

4.1.1. Requeriments funcionals generals

S'ha de dissenyar un formulari intel·ligent que sigui capaç de validar camps numèrics, que validi la lletra del DNI, que comprovi que els camps requerits estiguin omplerts, que porti incorporat el camp de la signatura digital, un codi de barres amb dos dimensions, per a la possible extracció de la informació mitjançant aquest.

L'usuari ha d'haver estat validat pel sistema abans de fer qualsevol acció.

L'administrador ha d'haver estat validat pel sistema abans de fer qualsevol acció.

L'usuari ha de ser capaç d' adjuntar un formulari al Servidor prèviament omplert.

L'aplicació en temps real, haurà d'extreure tota la informació que aquest porta associada al formulari adjuntat i inserir-la a la base de dades.

L'usuari també ha de ser capaç de descarregar-se el formulari intel·ligent amb dades prèviament omplertes .

L'administrador ha de ser capaç de consultar els formularis adjuntats pels ciutadans, i per a cada formulari, l'administrador ha de ser capaç de consultar les dades associades.

L'administrador ha de ser capaç d' extreure la informació mitjançant un codi de barres associat a un formulari.

4.1.2. Requeriments no funcionals

Usabilitat

L'aplicació ha de ser fàcil d'utilitzar. Qualsevol ciutadà amb uns mínims coneixements sobre Internet ha de ser capaç de realitzar totes les tasques de l'aplicació.

Eficiència

El temps de resposta per a un event interactiu de l'usuari, dependrà de la connexió, però amb una connexió bona, no haurà de ser superior a 3 segons.El refresc de pantalla dependrà de la connexió, però amb una connexió bona, no haurà de ser superior a 2 segons.

Robustesa

El sistema ha de ser tolerant amb determinats errors de l'usuari:

- Quan l'usuari no emplena bé el formulari.
- Quan l'usuari s'oblida emplenar algun camp.
- Quan l'usuari no es valida correctament.
- Quan l'usuari no adjunta cap document.

Implementació

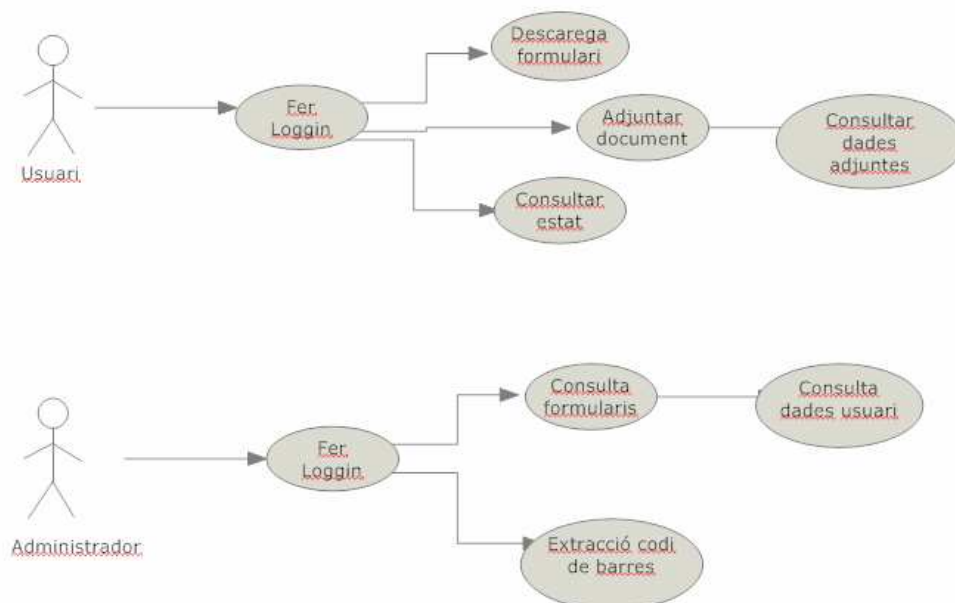
La validació dels formularis es farà en Javascript.

El disseny de l'aplicació es farà en Java.

El disseny del formulari serà amb l' Adobe LiveCycle Designer.

El procés d'importació i exportació de dades es farà a través de l'Adobe Form Server.

Diagrama Casos d'Ús



Diagrama

4.2. Fases projecte Adobe Forms

Disseny Formulari
Disseny Base de Dades
Disseny pàgina web

4.2.1. Disseny dels formularis

En aquesta part del projecte es pretén dissenyar un formulari intel·ligent, per la Sol·licitud de la prestació de Renda Bàsica d'Emancipació, que seguirà el model de la Generalitat de Catalunya. Per al disseny del formulari, utilitzarem la plataforma ADOBE FORMS, concretament l'eina d'Adobe LiveCycle Designer.

En aquest apartat, és important, determinar quina serà l'estructura del XML per les dades.

En el nostre cas, l'esquema generat és el següent:

```
<?xml version="1.0" encoding="UTF-8"?>
<xfa:datasets xmlns:xfa="http://www.xfa.org/schema/xfa-data/1.0/">
  <xfa:data>
    <form1>
      <page1>
        <part1>
          <nom> </nom>
          <cognoms> </cognoms>
          <data_naixament/>
          <RadioButtonList/>
          <dni></dni>
          <telefon_fix/>
          <telefon_mobil/>
          <seg_social/>
          <tipus_previsio/>
          <num_tip_previsio/>
          <sPosLaboral/>
          <nom_banc/>
          <entitat_banc/>
          <oficina_banc/>
          <DC_banc/>
```

```

        <compte_banc/>
        <altres/>
        <CheckBox1>0</CheckBox1>
        <email/>
    </part1>
    <banc_diferent>
        <nom_banc/>
        <entitat_banc/>
        <oficina_banc/>
        <DC_banc/>
        <compte_banc/>
    </banc_diferent>
    <adreces>
        <hola/>
        <tipus_via_AN/>
        <nom_AN/>
        <num_AN/>
        <bloc_AN/>
        <escala_AN/>
        <pis_via_LL/>
        <porta_via_LL/>
        <codi_postal_LL/>
        <poblacio_AN/>
        <sLloguer/>
    </adreces>
    <Subform1>
        <tipus_via_LL/>
        <nom_via_LL/>
        <num_AN1/>
        <bloc_AN1/>
        <escala_AN1/>
        <pis_via_LL1/>
        <porta_via_LL1/>
        <codi_postal_LL1/>
        <poblacio_AN1/>
    </Subform1>
    <staticfinal>
        <ref_cad_LL/>
    </staticfinal>

```

```

</page1>
<dades_bancaries_titular>
    <nom_banc_t/>
    <entitat_banc_t/>
    <oficina_banc_t/>
    <DC_banc_t/>
    <compte_banc_t/>
    <nif_titular/>
    <cognoms_titular/>
    <nom_titular/>
</dades_bancaries_titular>
<anterior>0</anterior>
<posterior>0</posterior>
<dosdeu_ant>0</dosdeu_ant>
<dosdeu_post>0</dosdeu_post>
<sis_post>0</sis_post>
<uvint_post>0</uvint_post>
<localitat_data/>
<dni_doc>0</dni_doc>
<sEspanyol/>
<contracte_lloguer>0</contracte_lloguer>
<DateTimeField1/>
<justific_ingressos>0</justific_ingressos>
<ha_treballat>0</ha_treballat>
<no_ha_treballat>0</no_ha_treballat>
<vida_laboral>0</vida_laboral>
<si_6m>0</si_6m>
<no_6m>0</no_6m>
<no_6m_doc>0</no_6m_doc>
<no_6m_doc1>0</no_6m_doc1>
<altres_doc>0</altres_doc>
<altres_doc1>0</altres_doc1>
<nom_cognoms/>
<nif/>
<declaracio1>0</declaracio1>
<declaracio2>0</declaracio2>
<declaracio3>0</declaracio3>
<declaracio4>0</declaracio4>
<declaracio5>0</declaracio5>
<declaracio6>0</declaracio6>
<declaracio7>0</declaracio7>

```

```
<declaracio8>0</declaracio8>
<declaracio9>0</declaracio9>
<declaracio10>0</declaracio10>
<declaracio11>0</declaracio11>
<localitat/>
<TextField1/>
<PDF417BarCode1>1234567890</PDF417BarCode1>
<DateTimeField1/>
</form1>
</xfa:data>
</xfa:datasets>
```


Com a resultat del disseny s'obtindrà una plantilla en format d'arxius amb extensió PDF. Aquest formulari té el camp de la signatura digital i duu incorporat un codi de barres en dues dimensions, de manera, que es podran extreure les dades passant aquest codi de barres per un escàner. El formulari també farà les corresponents validacions en Javascript, de forma, que les dades que introdueix l'usuari siguin correctes i també que els camps necessaris estiguin complets.

Concretament, les validacions que farà el formulari són el format del mail, que el DNI estigui correcte, és a dir, a través d'un petit script es comprovarà que la lletra correspongui amb el número del DNI introduït per l'usuari.

Aquest formulari també validarà el nombre de dígit a introduir, en cas que estiguem parlant d'un camp numèric a través d'expressions regulars, com és el cas de l'apartat bancari. (En el annexa de la memòria trobem adjuntat el codi que hi ha darrere del formulari, el llenguatge és una combinació entre JavaScript i FormCalc, que és el llenguatge propi per al disseny de formularis amb l'eina Adobe Designer)

També es crearan dos apartats dinàmics que només apareixeran en cas que un botó lògic estigui actiu. Concretament, en l'apartat "Número de compte bancari", trobarem un botó lògic on es pregunta si el número de compte on es vol rebre l' ajuda és diferent al número on s'efectuarà el pagament, si no és així, a la part inferior d'aquest apareixerà un apartat, per tal de que l'usuari pugui posar el número. El mateix passarà amb l'apartat "Dades de l'habitatge llogat per a destinar a domicili habitual", que només apareixerà en cas que el botó "És la mateixa adreça de lloguer?" de la part "Adreça de Notificació" estigui actiu.

A continuació, es mostren dos imatges on es veu reflectit l'aspecte que acabo de comentar.

Número de compte bancari a través del qual s'efectuarà el pagament del lloguer

Denominació de l'entitat bancària Entitat Oficina DC Compte

Número de compte bancari a través del qual es rebrà la prestació. Si és diferent de l'anterior indiqueu-ho aquí.
Aquest compte ha de ser de la mateixa entitat de crèdit a través de la qual es fa el pagament

☐ Número de compte bancari diferent

Adreça de notificació

És la mateixa que l'adreça de lloguer? ☐ Si ☒ No

Tipus de via (carrer, plaça..) Nom de la via

Número Bloc Escala Pis Porta Codi Postal Població

Formulari amb botó lògic "Número de compte bancari" sense activar

Número de compte bancari a través del qual s'efectuarà el pagament del lloguer

Denominació de l'entitat bancària Entitat Oficina DC Compte

Número de compte bancari a través del qual es rebra la prestació. Si és diferent de l'anterior indiqueu-ho aquí.
Aquest compte ha de ser de la mateixa entitat de crèdit a través de la qual es fa el pagament

☒ Número de compte bancari diferent

Denominació de l'entitat bancària Entitat Oficina DC Compte

Adreça de notificació

És la mateixa que l'adreça de lloguer? ☐ Sí ☒ No

Tipus de via (carrer, plaça..) Nom de la via

Número Bloc Escala Pis Porta Codi Postal Població

Formulari amb botó lògic "Número de compte bancari" actiu

El formulari també seguirà una lògica referent als botons que estan actius. El formulari comprovarà que els botons que l'usuari activa estiguin lògicament relacionats, és a dir, per exemple, en la part "SOL·LICITO", el formulari ha de comprovar que siguin coherents els botons que l'usuari activa, i en cas que no sigui així, ho corregeix.

El formulari també disposarà de missatges d'error, per avisar als usuaris quan algun camp no està correctament omplert.

En aquesta primera part, el formulari es podrà omplir de forma offline, en format PDF, o bé, també disposarà d'un botó per imprimir-lo, pels ciutadans que prefereixin seguir la metodologia tradicional.

Finalment, també dir que s'incorporarà un botó per netejar les dades, i deixar el formulari en blanc, d'aquesta manera també facilitarem la feina als usuaris, alhora de corregir les seves dades.

Aquest formulari proporcionat a l'usuari, portarà dades actualitzades prèviament omplertes, i s'encarregarà de fer les validacions corresponents amb JavaScript. També es validarà que els camps estiguin omplerts i que el botons d'opció segueixin una lògica coherent.

Com a resultat del disseny obtindrem una plantilla del formulari amb extensió PDF (Adobe Dynamic XML Form).

4.2.2. Disseny i configuració de la base de dades

La base de dades serà implementada en Accés.

Les taules que formen la base de dades són les següents:

ADMINISTRADORS

ADRECA_LLOGUER

ADRECA_NOTIFICACIO

DADES_TITULAR_LLOGUER

ESTAT

USUARIS

CIUTADANS

ADMINISTRADORS

Aquesta és la taula utilitzada per validar el nom d'administrador i la contrasenya d'aquest. Aquest procés es duu a terme quan l'administrador vol entrar dins de l'aplicació.

Clau primària: id

iD	Autonúmeric
Nom_formulari	String
Nom_sol·licitant	String

ADRECA_LLOGUER

En aquesta taula, s'insereixen les dades referents a l'adreça de lloguer on el sol·licitant anirà a viure i pel qual està demanant l'ajuda. Aquest procés d'inserció de dades té lloc, una vegada l'usuari adjunta el formulari, s'extreuen les dades i s'insereixen a la base de dades.

Clau Primària: Id

Clau forana: DNI

Id	Autonúmeric
DNI	String
Carrer	String
Número	String
Bloc	String
Escala	String
Pis	String
Porta	String
Codi_postal	String
Poblacio	String

ADRECA_NOTIFICACIO

En aquesta taula podrem trobar les dades on l'Administració ha d'enviar les notificacions per a cada usuari. Aquest procés d'inserció de dades té lloc, una vegada l'usuari adjunta el formulari, s'extreuen les dades i s'insereixen a la base de dades.

Clau Primària: Id

Clau forana: DNI

Id	Autonúmeric
DNI	String
Carrer	String
Número	String
Bloc	String
Escala	String
Pis	String
Porta	String
Codi_postal	String
Poblacio	String

DADES_TITULAR_LLOGUER

En aquesta taula trobarem les dades del titular del lloguer. Aquest procés d'inserció de dades té lloc, una vegada l'usuari adjunta el formulari, s'extreuen les dades i s'insereixen a la base de dades.

Clau primària: Id

Clau forana: DNI_sol·licitant

Dni_sol·licitant	String
Nom	String
Cognoms	String
dni	String
Num_bancari	String

ESTAT

Aquesta és la taula on l'aplicació consultarà l'estat de la seva sol·licitud. Aquesta consulta té lloc en el moment que l'usuari vol consultar l'estat de la tramitació de la seva sol·licitud.

Clau primària:Id

Clau forana: DNI

Id	Autonúmeric
usuari	String
estat	String
DNI	String

USUARIS

Taula utilitzada per a la validació del nom d'usuari i contrasenya. Aquest procés es duu a terme quan l'administrador vol entrar dins de l'aplicació.

Clau primària: Id

Clau forana: DNI

Id	Autonúmeric
Usuari	String
Password	String
DNI	String

CIUTADANS

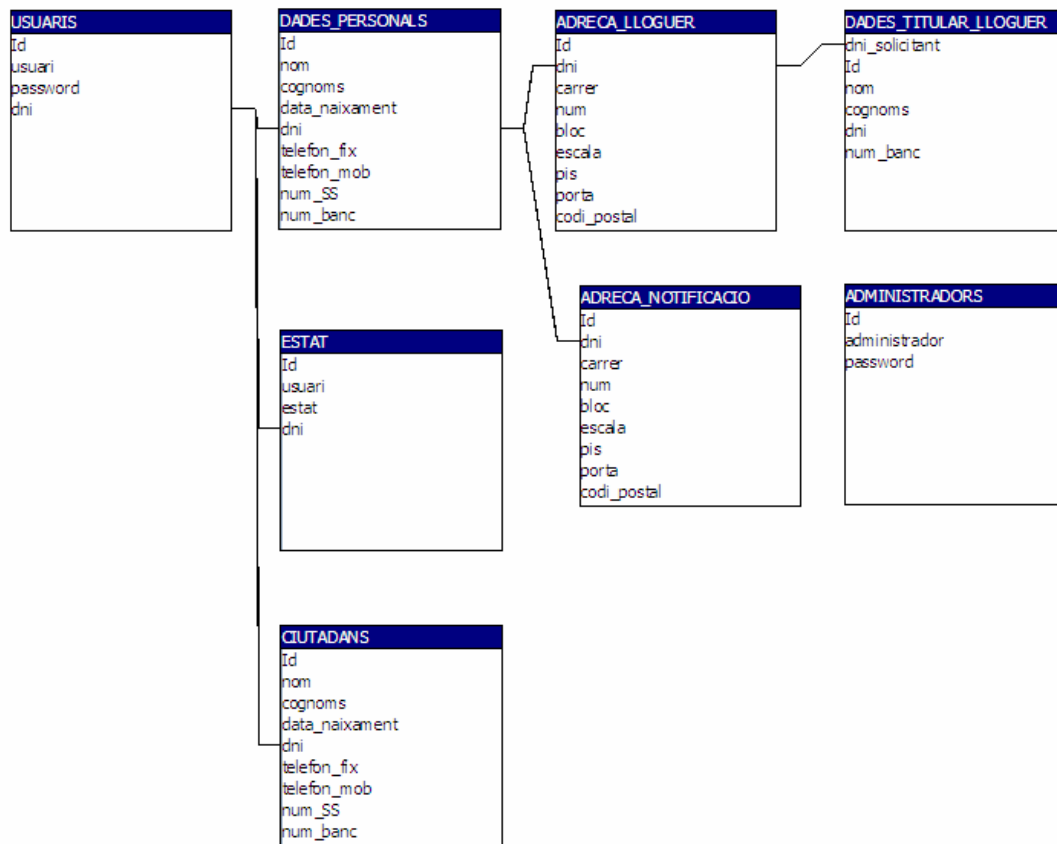
Aquesta taula representa on l'Administració té guardades totes les dades referents als ciutadans de Catalunya.

Clau primària: Id

Clau forana: DNI

iD	Autonúmeric
Nom	String
Cognoms	String
DNI	String

Informe Resultats Disseny Base de Dades



Informe

4.2.3. Disseny aplicació web

En aquesta pàgina web representarem el procés que simul·li un ciutadà de Catalunya que vol fer una sol·licitud per demanar una ajuda d'Emancipació. Per tant, el disseny de la pàgina web tindrà similitud a les pàgines de la Generalitat de Catalunya.

Crearem un jsp anomenat "mainLayout", encarregat de carregar-se cada vegada, on se li passarà un paràmetre que indicarà quin jsp ha de carregar-se en aquell moment. Podríem dir doncs, que una part de l'aplicació serà estàtica i únicament anirà canviant la part del mig.

En l'aplicació web són importants tres processos:

Descarrega del formulari

Per realitzar la sol·licitud del formulari, des de un portal web que representi a l'Administració, el ciutadà podrà descarregar-se un document PDF i l'aplicació generarà un formulari per a la sol·licitud de la prestació de la Renda Bàsica d'Emancipació.

Al realitzar la descarrega del formulari, es consultarà mitjançant una connexió a la base de dades, algunes dades actualitzades sobre l'usuari que està fent la petició. A continuació es crearà un arxiu XML amb el format del formulari, i les dades que acabem d'extreure de la base de dades i a partir d'aquí, aquesta informació s'inclourà dins del PDF, de manera que l'usuari podrà guardar el PDF en el seu ordinador, amb la informació d'aquests sistemes actualitzada en el moment de la descarrega. Per poder-ho fer, serà necessari estendre les propietats del PDF generat amb el Adobe LiveCycle Extensions. Amb aquest producte permetrem que amb l'Acrobat Reader puguem omplir els camps de text, guardar el document o inclús adjuntar documents en el propi PDF.

Una vegada l'usuari hagi omplert el document, el firmarà, es faran les validacions corresponents i el podrà adjuntar al Servidor de l'Administració Pública, o be, portar-lo a l'administració com s'havia fet tradicionalment.

Processament dels formularis

Una vegada s'hagi adjuntat el document a l'Administració, l'aplicació,

a través d'una connexió al Servidor Adobe, s'encarregarà de generar un fitxer XML, que conté tota la informació que ha inclòs el ciutadà de la sol·licitud.

Un cop tinguem aquest fitxer, parcejarem tot el arxiu de manera que guardarem en memòria tota la informació de l'usuari. A partir d'aquí, ja podrem guardar la informació a la base de dades per tal de procedir a la seva resolució.

Apart d'aquests dos processos destacats dins de l'aplicació l'usuari podrà consultar l'estat de la seva sol·licitud.

Dins de l'aplicació també podrem entrar com a administrador. L'administrador en un primer moment, podrà consultar els formularis adjunts i per a cada un d'ells, podrà consultar les dades que hi duu associades. També es pretén que l'administrador, pugui passar per un escàner el codi de barres que es troba en el formulari, per tal, que els ciutadans que prefereixin portar els papers per sol·licitar l'ajuda a l'Administració, puguin fer-ho i un moment allà, l'abstracció de dades es podria fer de forma directa, passant el codi per un escàner.

4.3. Entorn tecnològic

4.3.1. Entorn operatiu

Aplicació web

Servidor d'aplicacions: Tomcat v6.0

Client: Navegador Web amb els pluggins de Acrobat Reader (per exemple, Internet Explorer 5.0 o superior, Firefox) o el Adobe Acrobat Reader, sent necessaris una versió 7.0.8. com a mínim i un certificat digital per firmar els formularis.

Adobe LiveCycle

Servidor d'aplicacions: JBoss

En el servidor es troben instal·lats els productes d'Adobe LiveCycle que a continuació detallarem:

LiveCycle Forms ES 8.0.1

LiveCycle Reader Extensions ES 8.0.1

4.3.2. Entorn lògic de Programació

Adobe LiveCycle

Adobe LiveCycle Designer 8.1.1 és el programa amb el que es preparen els arxius PDF. Aquests arxius seran la plantilla PDF dels formularis.

Eclipse per al disseny de l'aplicació en Java.

4.3.3. Arquitectura

Es proposa com arquitectura per al desenvolupament, la següent:

Client: Navegador Web amb els plugings de Acrobat Reader o el Adobe Acrobat Reader, sent necessaris una versió 7.0.8 com a mínim i un certificat digital amb el que firmar els formularis.

Servidor Web: Tomcat 6.0

Servidor Aplicacions (aplicació): Tomcat v6.0

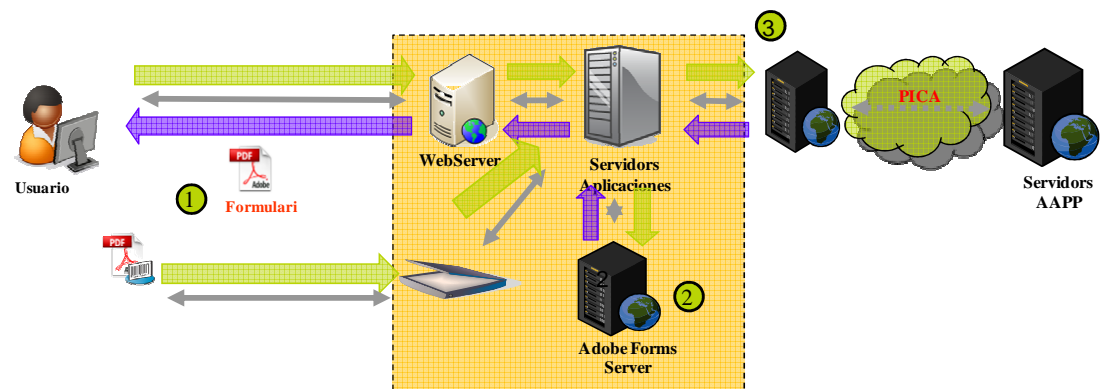
Servidor Aplicacions(LiveCycle): JBoss

Producte AdobeLiveCycle:

LiveCycle Forms ES 8.0.1. Aquest producte s'encarrega de generar arxius PDF a partir d'una plantilla predefinida. Aquestes plantilles es generen mitjançant l'Adobe LiveCycle Designer.

LiveCycle Reader Extensions ES 8.0.1. Aquest component és l'encarregat de donar permisos a un PDF, per tal de que un usuari pugui realitzar accions especials, com omplir el formulari o adjuntar arxius.

Detall arquitectura



Detall arquitectura Adobe Forms

4.4. Llenguatges utilitzats

4.4.1. HTML

Llenguatge de programació web estàtic. És el llenguatge web més estàndard, però té forces limitacions. El llenguatge funciona a través d'etiquetes i dades, i aquestes tenen un comportament diferent en quant a l'estructura per mostrar el contingut d'aquests. Una web HTML pura, només conté etiquetes i informació picada, per tant, no es té cap possibilitat per modificar la plana final.

4.4.2. CSS

Els fulls d'estils, o CSS, s'utilitzen per determinar la forma de presentació de les dades contingudes en les etiquetes HTML. La utilitat dels CSS és la separació entre la estructura de dades i la seva presentació, podent així canviar tot l'aspecte d'una plana sense modificar-ne el codi. Aquests fulls d'estil s'associen a la pàgina, i aquestes fan referència a les etiquetes segons unes classes, i a les classes es determina la forma de presentació del contingut.

4.4.3. Javascript

La programació Javascript complementa a la programació HTML. És tracta d'una programació interpretada i orientada a objectes, per tant, no requereix compilació. Les sentències Javascript es descarreguen junt a la plana web, i s'executen a client. Un cop descarregades, no pot interactuar amb el servidor.

4.4.4. SQL

SQL (Structured Query Language o Llenguatge d'interrogació estructurat) és un llenguatge estàndard de comunicació amb bases de dades relacionals. És a dir, un llenguatge normalitzat que permet treballar amb la majoria de bases de dades relacionals. L'SQL es pot utilitzar dins d'altres llenguatges de programació. La principal característica d'aquest llenguatge és la seua simplicitat, ja que amb pocs coneixements es poden fer consultes bàsiques sobre una base de dades, encara que no per això deixa de ser un llenguatge complet, tant relacionalment com computacionalment (a partir de la versió SQL3 publicada el 1999).

El llenguatge SQL es pot dividir en tres conjunts d'instruccions o sentències segons la seva funció:

Les sentències de definició (DDL Data Definition Language) permeten crear, modificar i esborrar estructures de dades.

Les sentències de manipulació (DML Data Manipulation Language) permeten llegir, inserir, modificar i esborrar registres.

Les sentències de control (DCL Data Control Language) permeten finalitzar o rebutjar transaccions i autoritzar o revocar els permisos dels usuaris.

L'SQL també estandarditza les sentències que s'utilitzen per hostatjar-se dins d'altres llenguatges.

4.4.5. Java

El llenguatge de programació Java fou dissenyat per James Gosling i els seus companys a Sun Microsystems, a l'any 1990, a partir del C++. Des del seu naixement fou pensat com un llenguatge orientat a objectes, és a dir, que segueix la filosofia de programar mòduls senzills, per tal de crear aplicacions avançades quan tots treballen junts.

Entre 13 de novembre de 2006 i el maig del 2007 Sun alliberà parts de Java com a programari lliure i de codi obert, usant la llicència GPL.

Aquest és un llenguatge interpretat i, per tant, pot semblar lent en comparació amb altres llenguatges, però ofereix un índex de reutilització del codi molt elevat i és possible de trobar moltes llibreries lliures de Java. És un llenguatge flexible i potent per la facilitat amb què es programa i els resultats que ofereix. Un dels trets que el caracteritza i que fa que sigui una eina molt valorada a l'hora de desenvolupar aplicacions distribuïdes, és el fet que sigui un llenguatge multi plataforma.

4.5. Descomposició en mòduls

Aquesta aplicació permet que el usuari d'Internet tingui la possibilitat de descarregar-se un formulari, amb algunes dades actualitzades i de registrar la seva sol·licitud a la base de dades.

Per l'agrupació lògica de les classes, s'utilitzen tres paquets:

4.5.1. Actions

En aquest paquet trobem les classes que defineixen les accions necessàries de l'aplicació.

Adjunta.class

Aquesta servlet s'encarrega de carregar la jsp corresponent per tal de que l'usuari pugui carregar el formulari corresponent. Concretament, s'encarrega de carregar "adjunta.jsp".

Adjunta_anonim.class

Aquesta servlet, li passa a l'atribut usuari el valor d'anònim per tal de carregar el jsp corresponent per aquest usuari. Concretament cargarà "l'adjunta.jsp" amb algun valor diferent a l'usuari normal.

Checking.class

Aquest servlet és l'encarregat de dirigir l'aplicació, un cop s'ha fet la validació contra la base de dades del nom d'usuari i contrasenya, d'aquest. Posteriorment carrega el "form.jsp".

Checking_admin.class

Aquest servlet s'encarrega de dirigir l'aplicació un cop l'administrador s'ha validat. El jsp que carrega és el "form_admin.jsp".

Consulta.class

Servlet encarregat de comprovar l'estat de la sol·licitud de l'usuari a la base de dades, assignar-li a la variable estat, el resultat de la queri i carregar el jsp, on aquest pot veure-ho per pantalla, concretament l' "estat.jsp".

DecodeCodiBarra.class

Servlet encarregat de cridar al procés per a l'extracció de la informació del codi de barres, que pròximament explicaré amb detall i de carregar el jsp on l'administrador pot consultar, quina informació s'ha extret d'aquest.

El jsp és l'"infocode.jsp".

Exportacio.class

Servlet encarregat de carregar el formulari del usuari al Servidor Tomcat, per tal de poder treballar amb ell, realitzant les accions necessàries, que pròximament descriure amb detall. Aquest servlet fa una crida a la classe ExportData(), JDOM(), InsertBD(), i finalment carregar el jsp, on l'usuari pot consultar les dades del formulari que ha adjuntat per tal de validar si són correctes, anomenat "form_exit.jsp".

Form_admin.class

Servlet que s'encarrega de dirigir l'aplicació al jsp on l'administrador pot consultar tots els formularis adjuntats pels ciutadans, anomenat "formularis_adjunts.jsp".

Form_anonim.class

Servlet encarregat de carregar el menú de l'usuari anònim. El jsp és el "form_anonim.jsp".

Form_user.class

Servlet que s'encarrega de dirigir l'aplicació al jsp on el ciutadà pot consultar el seu menú d'usuari, anomenat "form.jsp".

Importacio.class

Servlet encarregat de dirigir l'aplicació a les classes necessàries per tal de que de proporcionar-li a l'usuari un formulari prèviament carregat amb les dades actualitzades de l'usuari que està realitzant la sol·licitud. Aquest servlet crida a les classes (que més tard descriure en detall) CrearXML(), ImportData(), i finalment carrega el jsp, on l'usuari té un link per descarregar-se el formulari, anomenat "pdf_generat.jsp".

Index_admin.class

Servlet encarregat de carregar la pàgina on l'administrador posa el seu nom d'usuari i la seva contrasenya, aquest és l' "Index_admin.jsp".

Index_user.class

Servlet encarregat de carregar la pàgina on l'usuari posa el seu nom d'usuari i la seva contrasenya, aquest és l' "index_user.jsp".

Info_form.class

Servlet encarregat de carregar la pàgina on l'administrador pot consultar la informació dels diferents usuaris que han adjuntat documents. Abans de carregar aquest jsp, cridarà a la classe "ConsultaPersona", encarregada d'extreure la informació de la base de dades.

Inici.class

Servlet encarregat de carregar la pàgina inicial de l'aplicació.

4.5.2. **Interactions**

ArticuloDAO.class

Aquesta classe implementa el patró DAO encarregat de realitzar el treball d'accés a la base de dades.

ConsultaPersona.class

Classe formada pel mètode Consulta().

Consulta()

Paràmetres d'entrada: String id

Funcionalitat: Aquesta classe s'encarrega de connectar-se a la base de dades, i extreure la informació sobre un usuari, en un funció de la seva id per realitzar els corresponents sets a l'objecte persona amb l'objectiu de mostrar-li al administrador per pantalla. Aquest mètode retorna un objecte persona.

Formularis.class

Classe formada pel mètode getfilas()

Getfilas()

Paràmetres d'entrada: No té cap paràmetre d'entrada

Funcionalitat: encarregat de connectar-se a la base de dades i extreure en un array els diferents formularis adjuntats pels usuaris per tal de poder-los imprimir per pantalla. D'aquesta manera l'usuari pot fer-se una idea de les diferents sol·licituds que han fet els ciutadans, i posteriorment consultar les dades d'aquests. Aquest mètode retorna un array amb la corresponent informació de cada formulari.

Check.class

Classe encarregada de validar contra la base de dades el nom d'usuari

i la contrasenya dels ciutadans i dels administradors
Aquesta classe està formada per dos mètodes.

Validar_usuari()

Paràmetres d'entrada: String username, String password

Funcionalitat: Aquest mètode s'encarrega de, un cop connectats a la base de dades, comprovar que el nom d'usuari i la contrasenya siguin correctes per tal de procedir amb l'aplicació.

Validar_admin()

Paràmetres d'entrada: String adminame, String password

Funcionalitat: Aquest mètode s'encarrega de, un cop connectats a la base de dades, comprovar que el nom d'administrador i la contrasenya siguin correctes per tal de procedir amb l'aplicació.

ExportData.class

Classe formada pel mètode matchXMLPDF().

matchXMLPDF()

Paràmetres d'entrada: String inputPDF, String outputXML

Funcionalitat: Aquest mètode es connecta al servidor Adobe i amb el pdf proporcionat per l'usuari (la ruta del qual es troba dins de la variable inputPDF) s'extreuen les dades, i es crea un XML, per poder tal de poder treballar amb aquestes dades. Aquest XML serà guardat a la ruta que li indica la variable outputXML.

InsertBD.class

Classe amb el mètode Insert.

Insert()

Paràmetres d'entrada: objecte persona

Funcionalitat: Aquesta funció s'encarrega de carregar a la base de dades, totes les dades extretes del formulari, en la seva corresponent taula. Aquest mètode retorna l'objecte persona.

JDOM.class

Classe amb el metode parser().

Parser()

Paràmetres dentrada: String inputXML

Funcionalitat: Aquesta funció s'encarrega, a partir del XML que troba a la ruta proporcionada pel paràmetre d'entrada inputXML de parcejar-lo. Recorrerem tot el fitxer, i retornarem un objecte persona amb els seus atributs corresponents.

ImportData.class

Classe que s'encarrega de generar un pdf, amb les dades actualitzades del usuari que esta fent la sol·licitud per descarregar-se el formulari.

Aquesta classe està formada per tres mètodes:

CrearXML()

Paràmetres d'entrada: string username

Funcionalitat: Aquest mètode s'encarrega de crear el fitxer xml, amb les dades corresponents de l'usuari a través d'una connexió a la base de dades. La ruta del fitxer xml, serà un parametre d'entrada per cridar a la classe que a partir del formulari en blanc i el fitxer XML, generarà el formulari que l'usuari es descarregarà posteriorment.

matchXMLPDF()

Paràmetres d'entrada: string inputXML, string inputPDF, string outputPDF

Funcionalitat: Aquest mètode s'encarrega, un cop connectat al Servidor Adobe, a través del formulari en blanc (la ruta esta dins la variable inputPDF) i el fitxer XML (inputXML), amb dades actualitzades de l'usuari de crear el PDF que serà proporcionat al client. El PDF creat es guardarà dins la variable outputPDF.

CodiBarraParser.class

Classe que s'encarrega d'extreure la informació del fitxer XML creat després de descodificar la informació del codi de barres.

Aquesta classe està formada per dos mètodes:

ParserXML()

Paràmetres d'entrada: string outputXML

Funcionalitat: Aquest mètode s'encarrega, a través del fitxer XML proporcionat en el paràmetre d'entrada, extreure la part on es troba el contingut que ens interessa. Concretament, en el tag "content", trobarem tota la informació que fa referència al codi de barres.

ParserContent ()

Paràmetres d'entrada: string content

Funcionalitat: Aquest mètode s'encarrega, a través del string proporcionat en el paràmetre d'entrada, parcejar-lo per extraure la informació personal de l'usuari.

DecodeFormData.class

Classe que s'encarrega d'extreure la informació codificada en el codi de barres i de crear un fitxer XML amb aquesta informació.

Aquesta classe està formada per dos mètodes:

DecodeForm ()

Paràmetres d'entrada: String inputPDF, string outputXML

Funcionalitat: Aquest mètode s'encarrega, a través del codi de barres proporcionat en el PDF, que prèviament ha estat escanejat, de generar un fitxer XML, amb l' informació que aquest codi porta associada.

convertDOM ()

Paràmetres d'entrada: org.w3c.dom.Document doc

Funcionalitat: Aquest mètode s'encarrega, de convertir l' objecte org.w3c.dom.Document en com.adobe.idp.Document.

4.5.3. Beans

Adreca_lloguer.class

Objecte que conté les variables, corresponents a l'adreça de lloguer del sol·licitant, és a dir, conté les dades de l'adreça que el ciutadà vol llogar i per el qual, sol·licita l'ajuda; i a través dels mètodes `get()` i `set()`, s'encarrega de assignar i recollir el valor de les variables. Les variables que pertanyen a l'adreça de lloguer són: carrer, número, bloc, escala, pis, porta, codi postal i població.

Adreca_Notificacio.class

Objecte que conté les variables corresponents a l'adreça de notificació, aquesta adreça correspon al lloc on el ciutadà vol rebre, la informació sobre la seva sol·licitud, aquesta pot coincidir amb l'adreça de lloguer. A través dels mètodes `get()` i `set()`, s'assignen i es recullen els valors d'aquests. Les variables que pertanyen a l'adreça de notificació són les mateixes que en el cas anterior, és a dir, carrer, número, bloc, escala, pis, porta, codi postal i població.

Dades_personals.class

Objecte que conté les variables corresponents a les dades personals de l'usuari. Aquestes variables són : nom, cognoms, data naixement, DNI, telèfon fix, telèfon mòbil, número de la Seguretat Social, número bancari, mail i número bancari alternatiu.

Dades_titular_lloguer.class

Objecte que conté les dades referent a la persona titular del lloguer de l'habitatge que vol fer l'usuari, el qual, esta demanant la sol·licitud. Aquestes dades són el nom, els cognoms, el DNI i el número bancari d'aquest.

Formulari.class

Objecte utilitzat per descriure el tipus d'ajuda que ha sol·licitat el ciutadà, i també el nom d'aquest. Les variables que formen aquest objecte són: nom del formulari i nom del sol·licitant, i a través dels mètodes `get()` i `set()`, recollim les dades per mostrar al administrador, els diferents formularis adjuntats pels ciutadans.

Persona.class

Objecte format pels quatre objectes descrits anteriorment: Dades Personals, Adreça de Lloguer, Adreça de Notificació i Dades Titular del Lloguer, aquestes quatre classes formen un objecte persona, el qual utilitzarem per passar com a paràmetre entre classes.

CodiBarra.class

Objecte que conté les variables corresponents a la informació que porta associada un codi de barres. Aquestes variables són : nom, cognoms, data naixement, DNI, telèfon fix, telèfon mòbil i mail.

4.5.4. Paràmetres

Definició dels paràmetres de la classe beans:

La classe Persona() formada per les 4 classes que a continuació descriure:

Dades Personals()

Paràmetre	Descripció	Tipus	Obligatori?
nom	nom usuari	string	Si
cognoms	cognoms usuari	string	Si
data_naixament	data naixament	string	Si
dni	dni	string	Si
telefon_fix	telefon fix	string	No
telefon_mob	Telefon mòbil	string	No
num_SS	Número seguretat social	string	No
num_banc	Número compte corrent	string	No
email	email	string	No
num_banc_dif	Número compte corrent alternatiu	string	No

Adreca_lloguer()

Paràmetre	Descripció	Tipus	Obligatori?
carrer	Nom carrer	string	Si
num	Número	string	Si
bloc	Bloc	string	No
escala	Escala	string	No
pis	Pis	string	No
porta	Porta	string	No
codi_postal	Codi postal	string	Si
poblacio	poblacio	string	Si

DadesTitularLloguer()

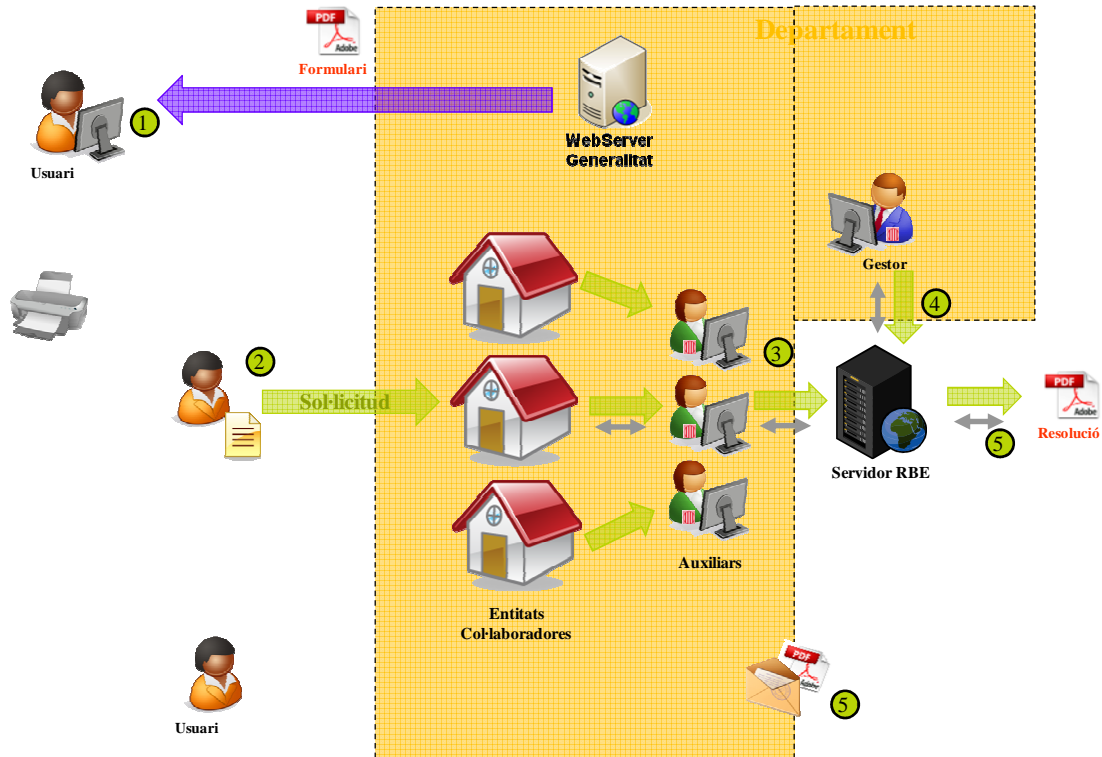
Paràmetre	Descripció	Tipus	Obligatori?
nom	Nom titular pis	string	Si
cognoms	Cognoms titular	string	Si
dni	Dni titular	string	Si
num_banc	Número compte	string	no

Adreca_notificacio ()

Paràmetre	Descripció	Tipus	Obligatori?
carrer	Nom carrer	string	No
num	Número	string	No
bloc	Bloc	string	No
escala	Escala	string	No
pis	Pis	string	No
porta	Porta	string	No
codi_postal	Codi postal	string	No
poblacio	poblacio	string	No

4.6. Representació visual Projecte Adobe Forms

4.6.1. Situació Actual

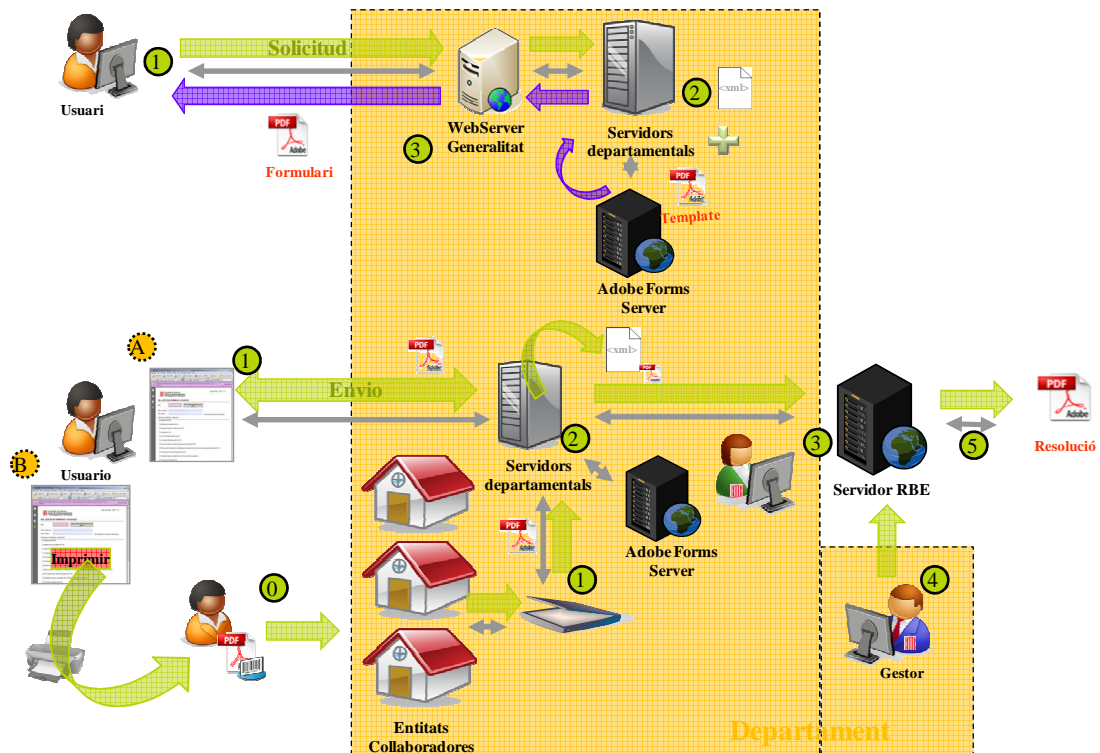


Situació actual

En aquesta imatge es vol representar, com actualment es duu a terme la tramitació d'Ajudes amb l'Administració.

Com podem observar, avui en dia, els ciutadans de Catalunya es descarreguen el formulari del Servidor, l'omplen de forma offline, se'l imprimeixen i un cop el tenen en paper, el poden dur a l'Administració, d'aquesta manera els auxiliars que treballen per l'Administració llegeixen les dades i les passen de forma manual al servidor per tal de procedir a la seva resolució. La resolució es entregada mitjançant una carta al seu domicili. Com podem deduir, d'aquesta manera el ciutadà es veu obligat a desplaçar-se fins a l'Administració per demanar l'ajuda i allà, es necessari que estiguin auxiliars especialment dedicats a passar les dades per que es pugui dur a terme la seva tramitació.

4.6.2. Situació desitjada



Situació desitjada

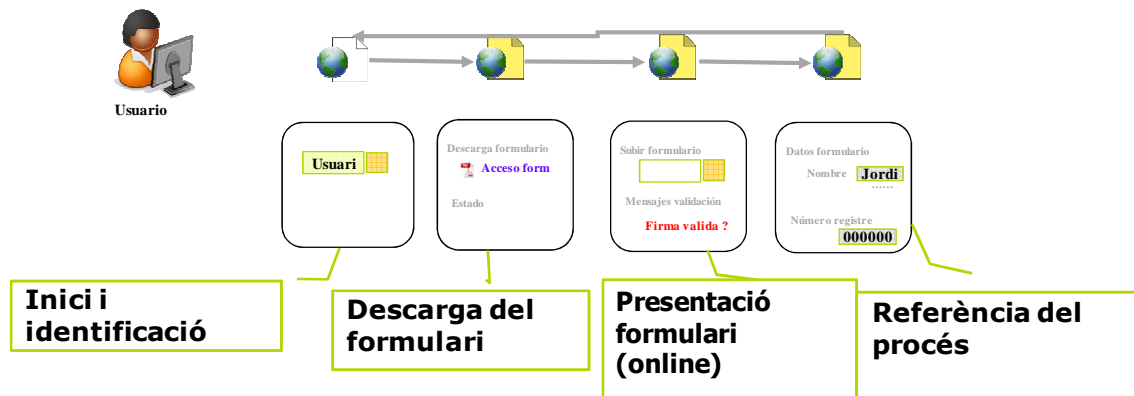
El que nosaltres volem és que l'usuari es pugui descarregar el formulari del Servidor de la Generalitat, i un cop el tingui en local, tingui dos opcions:

La prima opció consisteix en omplir en local i un cop el tingui acabat, el pugui adjuntar al Servidor, per tal que, la informació que porti associada sigui extreta en temps real i inserida a la base de dades, per procedir a la seva resolució.

La segona opció consisteix en que, un cop l'usuari hagi omplert el formulari, l'imprimeixi amb un codi de barres associat. Si aquest ciutadà prefereix portar aquest a l'administració, allà hi haurà una persona encarregada de passar el codi de barres per un escàner i les dades seran extretes al instant i inserides a la base de dades per procedir a la seva resolució.

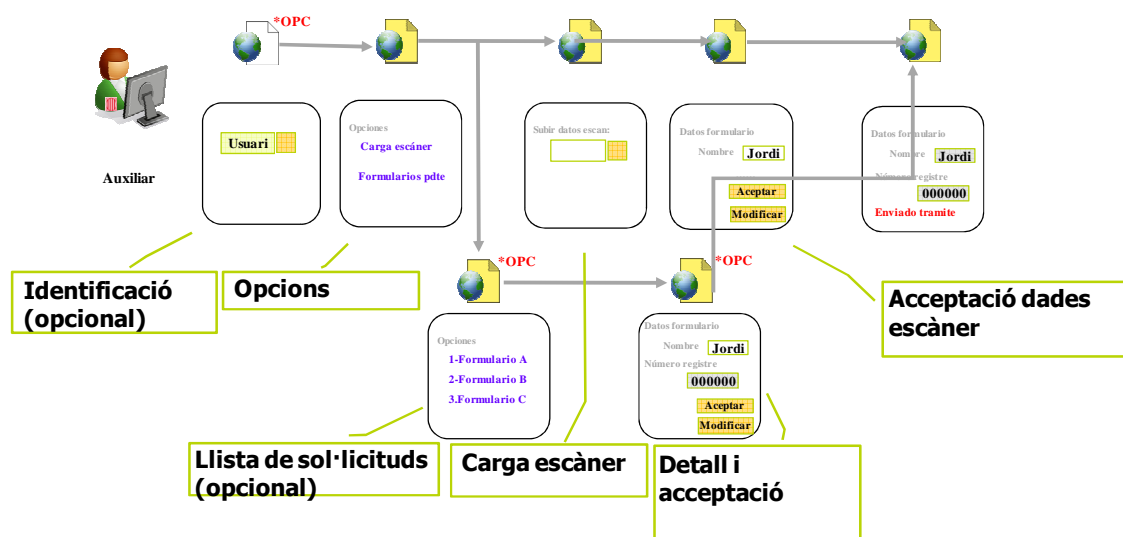
4.7. Descripció del portal WEB

Esquema general portal web des de el punt de vist de l'usuari:



Esquema general vist des de l'usuari

Esquema general portal web des de el punt de vist de l'administrador:

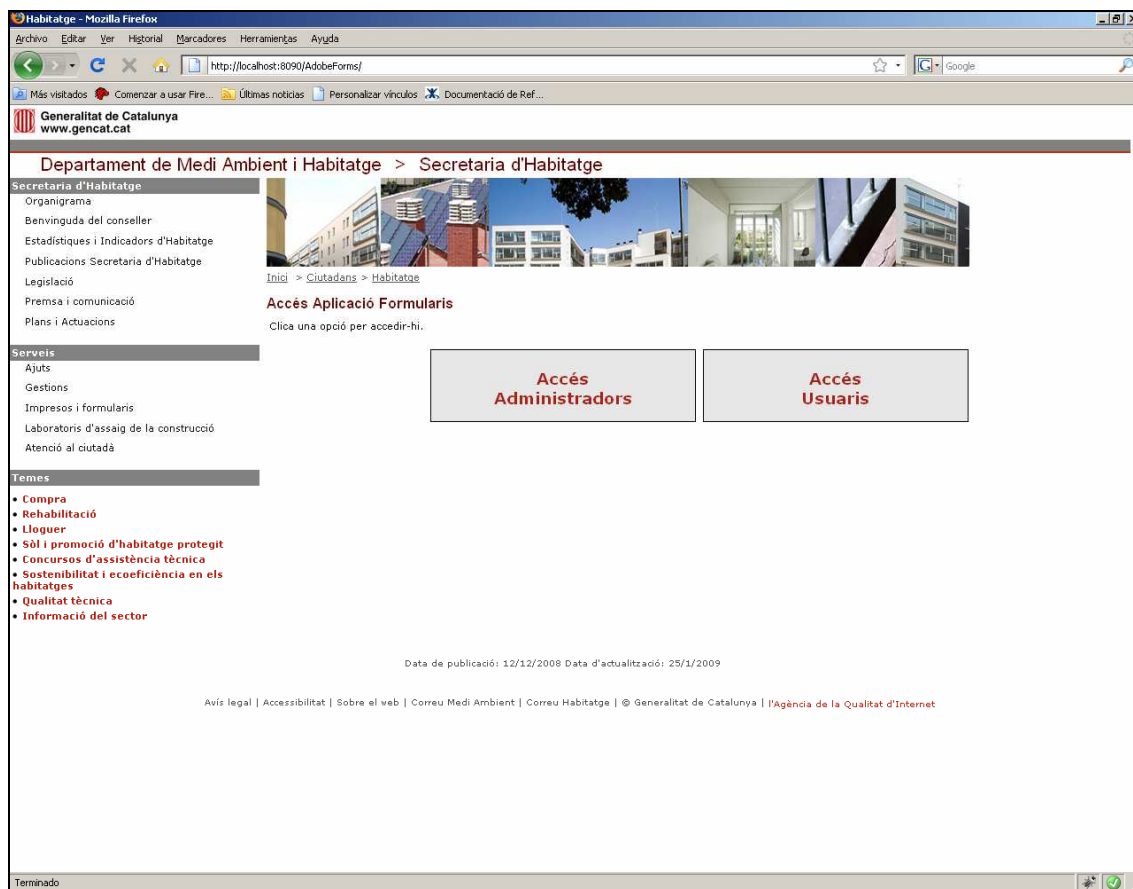


Esquema general vist des de l'Administrador

4.8. Portal WEB

En aquest apartat, descriurem el funcionament del portal Web.

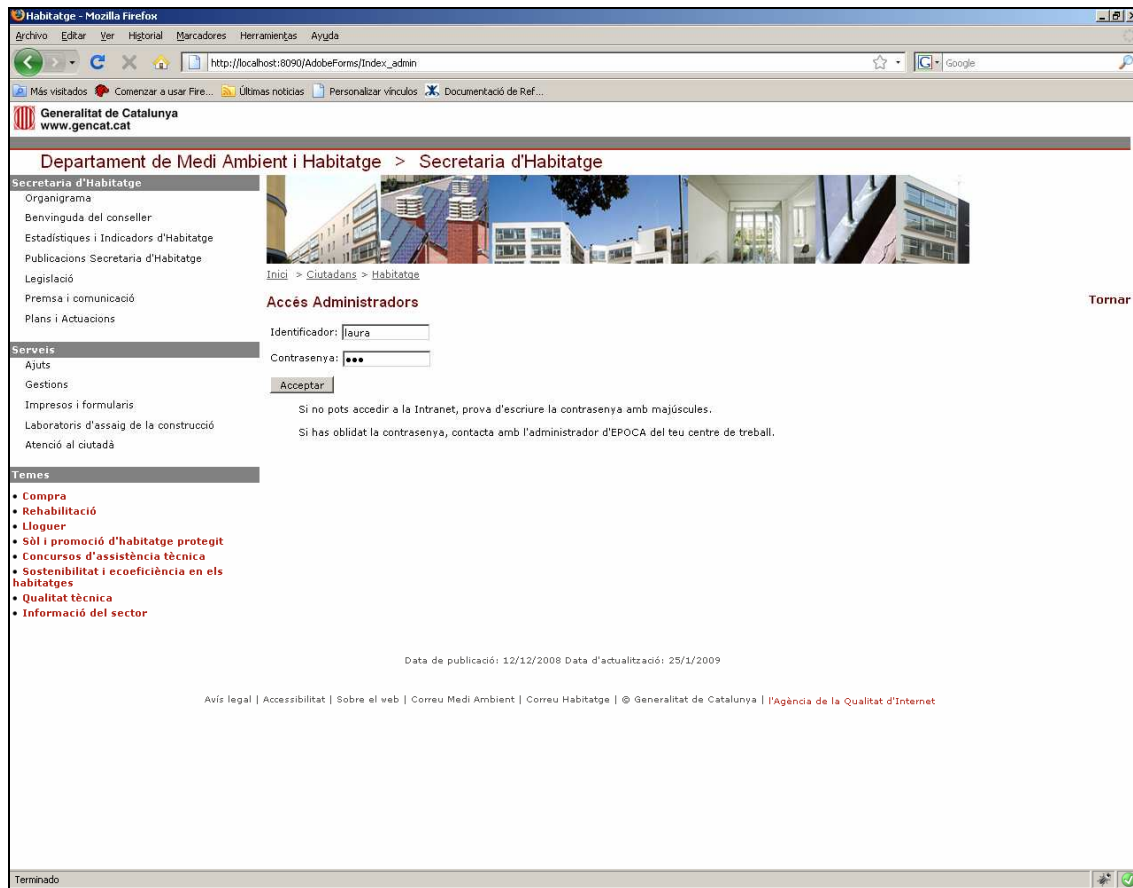
Per començar, al entrar a la pàgina ens trobem amb dos opcions, accedir com administrador i accedir com a usuari.



Index.jsp

4.8.1. Accés Administrador

Si accedim com a administradors, tal com es mostra a la següent pantalla, primer de tot ens haurem de validar com administradors.

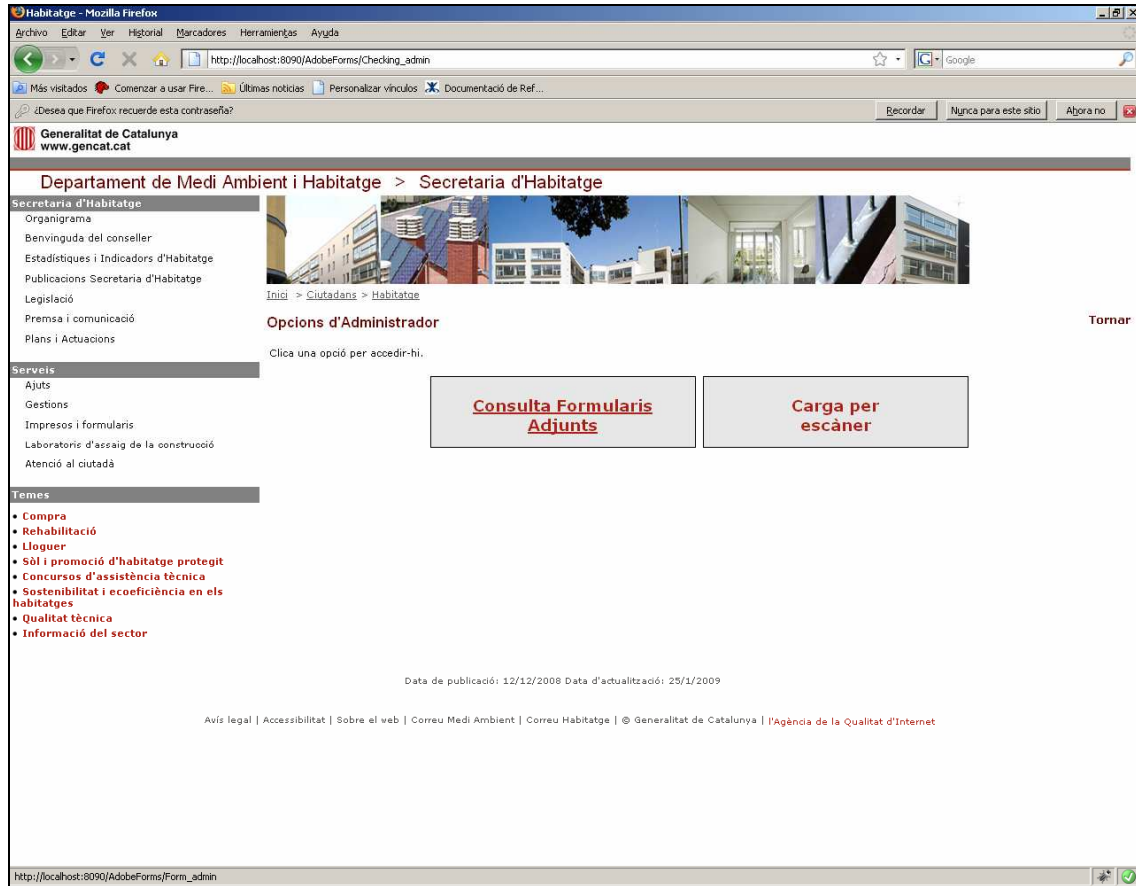


Index_admin.jsp

Si ens validem correctament, l'administrador tindrà dos possibilitats:

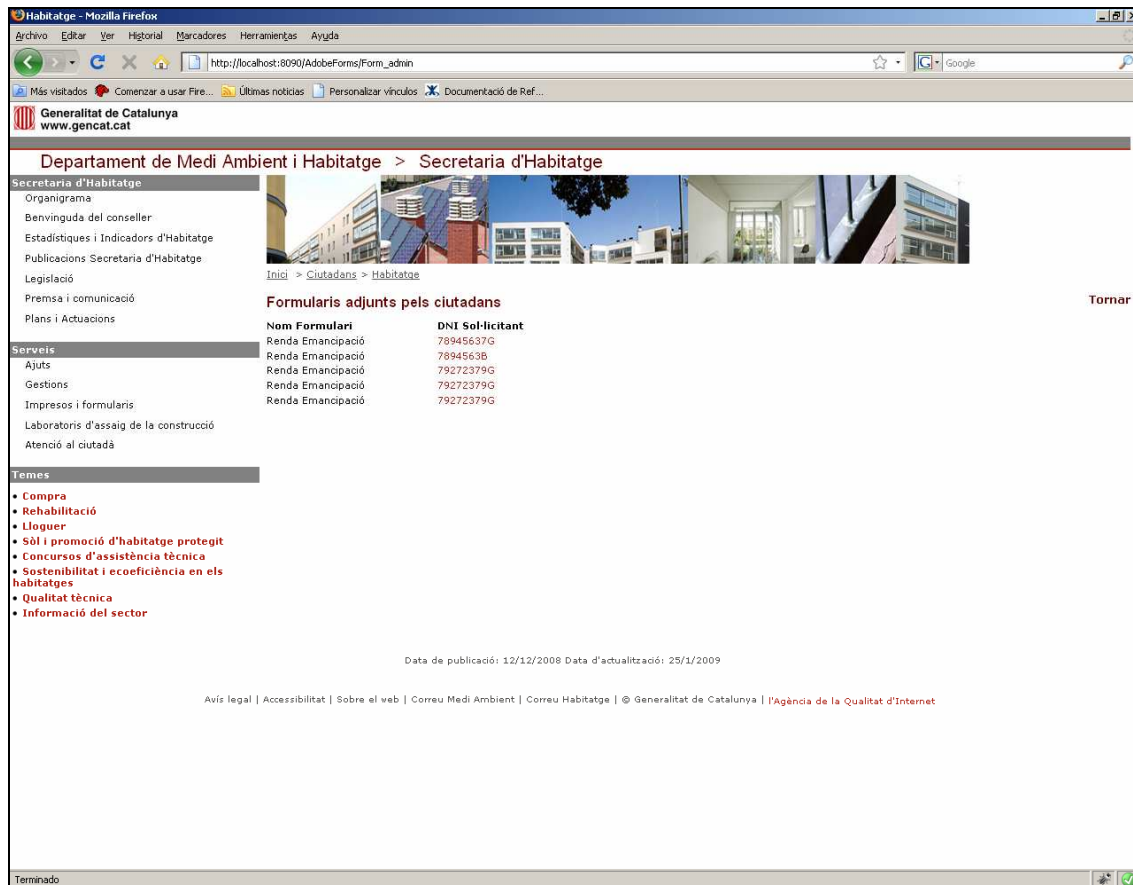
- Consultar els formularis adjunts pels ciutadans, de manera que, l'administrador pot tenir una idea, de les persones que han demanat aquesta ajuda
- Carga del formulari mitjançant l'escàner per extreure les dades. Aquesta opció permet, als ciutadans que ho prefereixin, tenir la possibilitat de dur el formulari a l'Administració (com es fa tradicionalment) i des d' allà l'administrador pot extreure les dades de forma automàtica mitjançant el codi de barres.

En aquesta pantalla es pot observar el menú de l'administrador:



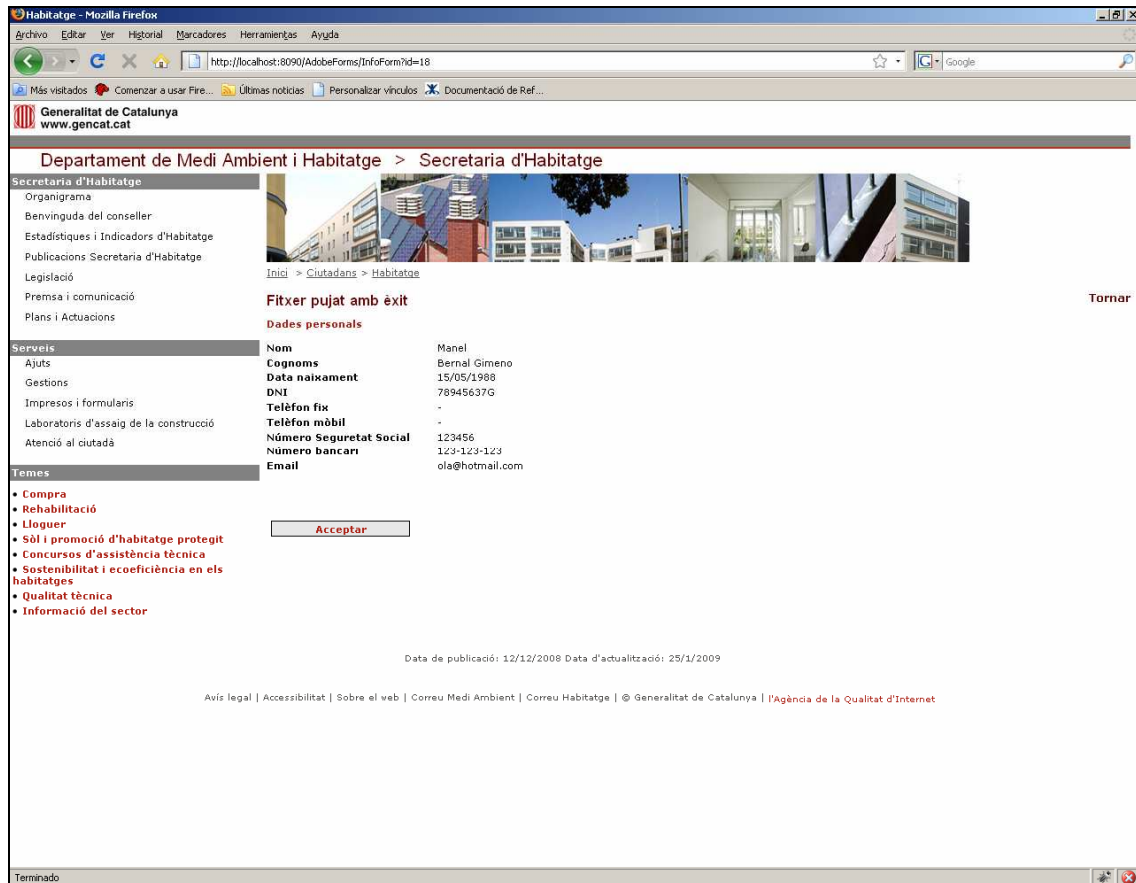
Form_admin.jsp

La web on es mostren els diferents formularis adjunts pels ciutadans:



Formularis_adjunts.jsp

I en cada formulari, podrem observar quina informació hi ha al darrere. És a dir, l'administrador, es capaç d'accedir a les dades que hi ha al darrere de cada formulari

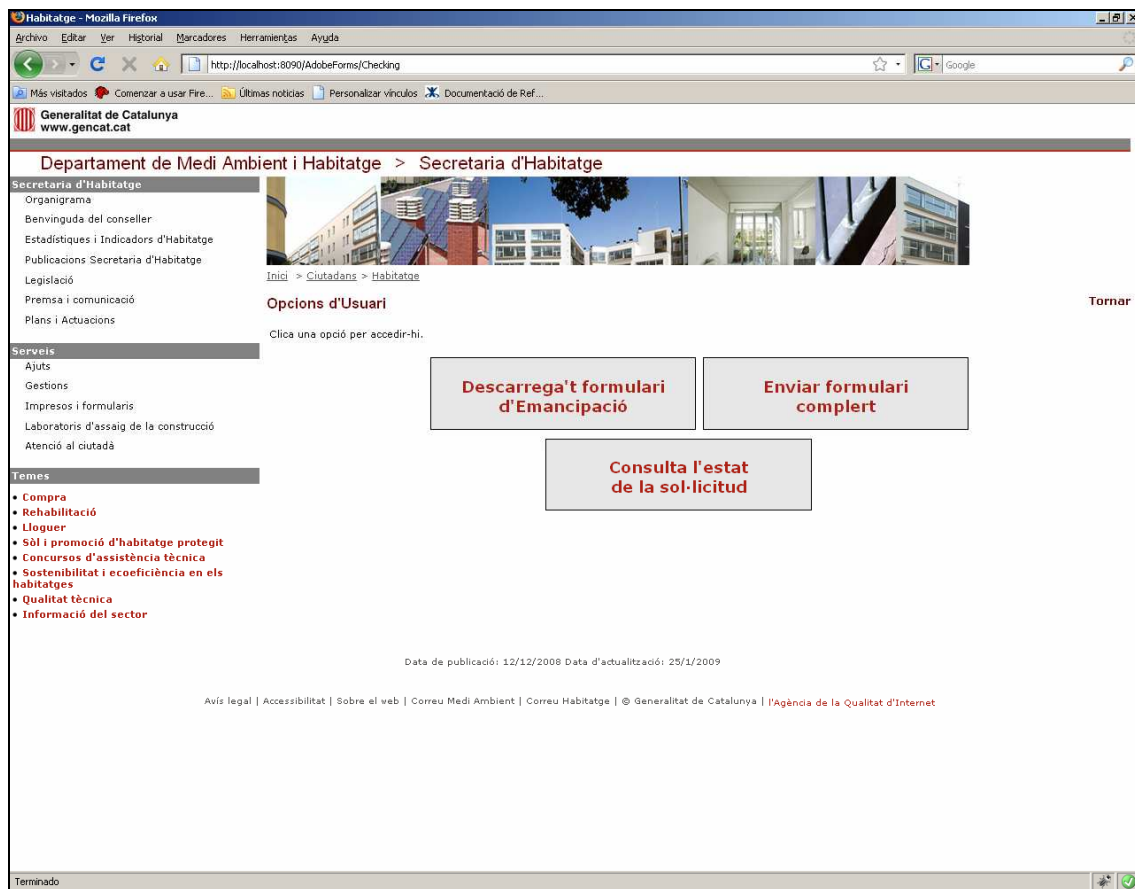


Info_user.jsp

4.8.2. Accés Usuaris

Pel que fa a l'usuari, un cop s'ha validat (la pantalla és la mateixa que en el cas de l'administrador). Tenim tres possibilitats diferents:

- Descarregar-se el formulari prèviament omplert.
- Enviar el formulari amb les dades de la persona que demana la sol·licitud.
- Consultar l'estat de la Sol·licitud.



Form.jsp

Si l'usuari es descarrega el formulari, la web et proporciona, el formulari amb dades prèviament omplertes del propi usuari.

En aquesta imatge podem observar, una part del formulari que se li proporciona a l'usuari. Aquest podrà omplir-lo i imprimir-lo per portar a l'Administració, on s'extrauran les dades manualment. Tot i que com a possible ampliació, el formulari portaria adjunt un codi de barres amb dos dimensions, amb totes les dades del formulari.

Generalitat de Catalunya
Departament de Medi Ambient i Habitatge
Secretaria d'Habitatge

Sol·licitud de la prestació de Renda Bàsica d'Emancipació

Dades de la persona titular del contracte de lloguer

Nom Cognoms Data de naixement Sexe ☐ H ☐ D

Numero de la Seguretat Social: - tipus de previsió social

En cas de no estar en el Règim General de la Seguretat Social: - número :

DNI Telèfon fix Telèfon mòbil Adreça correu electrònic

Situació laboral: ☐ Actiu/va per compte d'altri ☐ Autònom/a ☐ Pensionista Altres:

Número de compte bancari a través del qual s'efectuarà el pagament del lloguer

Denominació de l'entitat bancària Entitat Oficina DC Compte

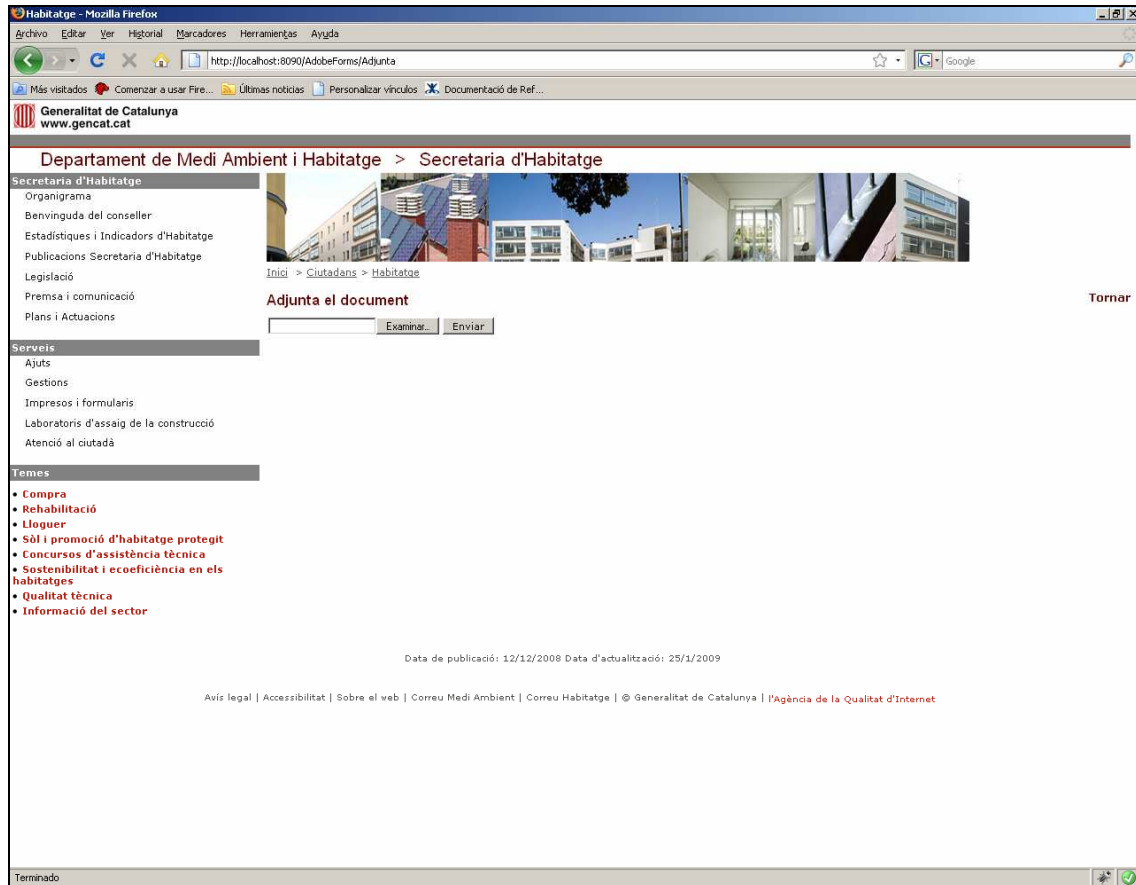
Número de compte bancari a través del qual es rebrà la prestació. Si és diferent de l'anterior indiqueu-ho aquí.
 Aquest compte ha de ser de la mateixa entitat de crèdit a través de la qual es fa el pagament

☐ Número de compte bancari diferent

Formulari Adobe

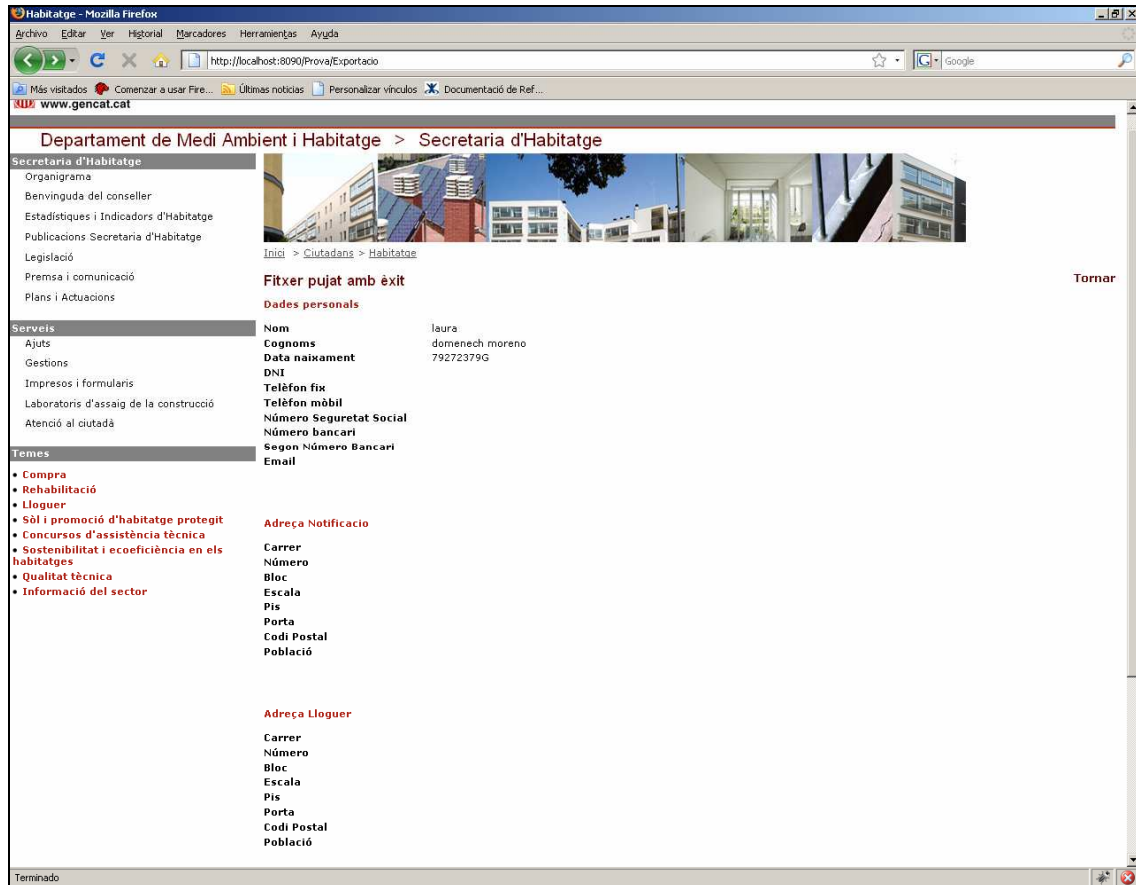
La segona opció que té l'usuari, és adjuntar el formulari per demanar l'ajuda. En una primera part, ens trobaríem amb la pantalla per a adjuntar el document. I tot seguit, la web ens mostrarà les dades introduïdes en el formulari, per tal de que l'usuari, tingui la possibilitat de comprovar que les dades introduïdes són correctes.

Pantalla per adjuntar el document:



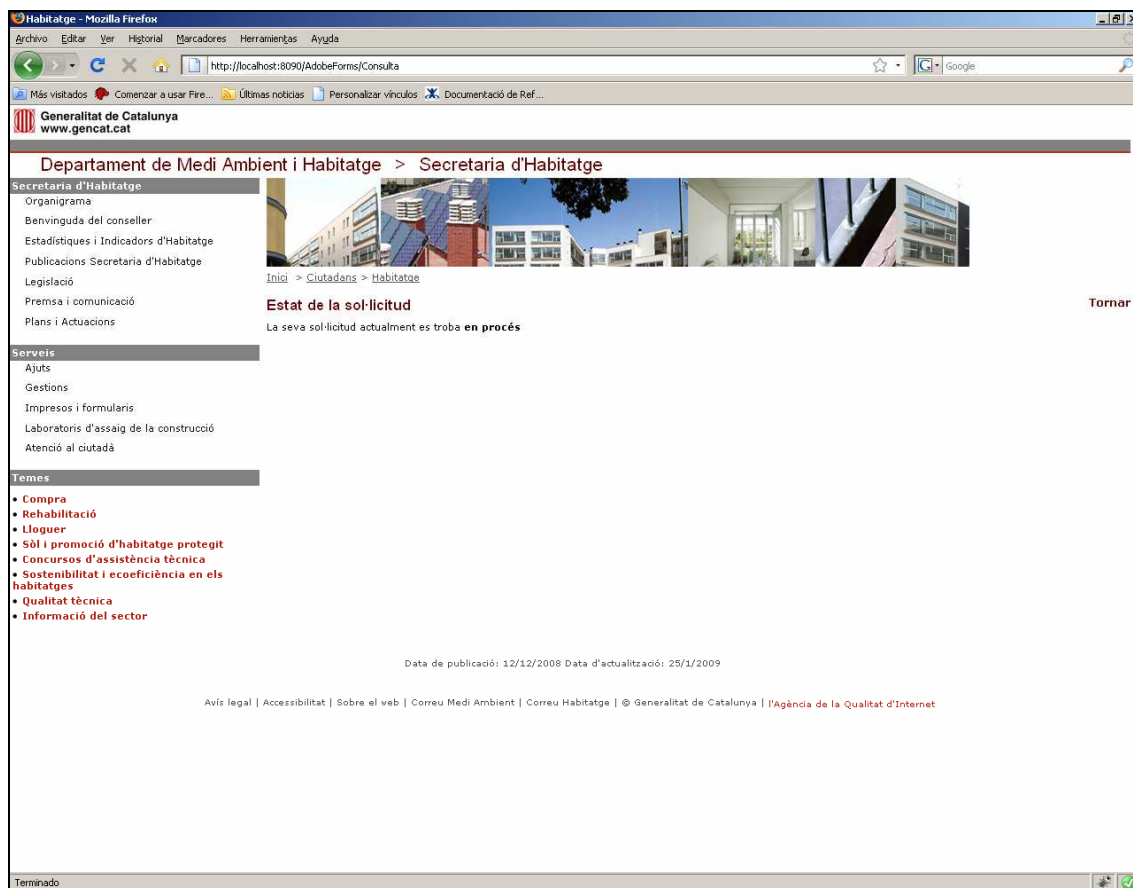
Adjunta.jsp

Pantalla on es mostren les dades:



Form_exit.jsp

Finalment, l'usuari també pot comprovar l'estat de la seva sol·licitud.



Estat.jsp

5. Conclusions

La creació d'aquest projecte ha suposat per a mi un gran esforç i un gran repte, ja que al iniciar el projecte, només tenia unes nocions bàsiques de llenguatge Java.

Professionalment, el haver treballat en una empresa fent el projecte ha significat un primer contacte amb el món laboral. Això m'ha permès fer-me una idea del funcionament de les empreses en aquest sector.

El projecte m'ha permès posar en pràctica tot una sèrie de coneixements adquirits durant la carrera d'Enginyeria Tècnica d'Informàtica de Gestió. La implementació dels Servlets de Java era un tema totalment desconegut per mi, tant en el tema de la codificació com la manipulació de les interfícies de programació i el software necessari. També per mi era totalment desconeguda la plataforma LiveCycle. Aquest són els punts on he emprat més temps del projecte per adquirir els coneixements necessaris per a poder realitzar l'aplicació.

Els objectius aconseguits són la importació de dades, donat un PDF en blanc, l'aplicació es capaç de omplir aquest PDF amb dades de l'usuari que actualment esta fent la sol·licitud del formulari. També he aconseguit donat un formulari PDF prèviament omplert per l'usuari, exportar les dades, de manera que sigui actualitzada al moment la base de dades. Es necessari dir, que el formulari que adquireix l'usuari és un formulari intel·ligent, és a dir, en temps real s'encarrega de fer les validacions amb Javascript necessàries i també de que l'usuari ompli tots els camps indispensables. L'últim objectiu ha aconseguit ha sigut la lectura del codi de barres del formulari, és a dir, al passar el codi de barres per un escàner, l'aplicació es capaç d'extraure tota la informació que duu incorporada per tal de procedir a la inserció a la base de dades

Tanmateix, l'aplicació esta dotada d'un mini portal web capaç de simular el procés que un ciutadà pot fer alhora de descarregar-se un formulari per demanar una sol·licitud a l'Administració.

Finalment dir, que aquest projecte m'ha permès adquirir coneixements sobre el disseny de formularis amb el Adobe Designer, el funcionament de la plataforma Adobe LiveCycle i la creació de pàgines web amb Java, és a dir, aspectes amb els quals mai havia treballat i que m'han ajudat a enriquir-me professionalment.

6. Futures línees d'actuació

Com a futures línees d'actuació, volem que l'aplicació, apart de llegir el codi de barres PDF417 sigui capaç de llegir altres tipus de codis de barres que el formulari pugui portar incorporat.

També ens interessa que l'usuari juntament amb el formulari pugui adjuntar documents, tal com el DNI, el passaport o documents similars. Per tal de fer això, alhora de l'extracció de dades es generarà un fitxer XDP (XML Data Package), amb les dades d'aquest i els documents adjuntats.

Com a propera línea d'actuació també volem que el formulari porti incorporat el camp de la signatura digital i l'aplicació sigui capaç de validar en temps real que la persona sigui realment la que diu ser.

També ens interessaria ser capaços de crear fluxos de treball, ja que, el Adobe Workbench permet realitzar aquesta tasca. Però aquesta tasca es realitzarà un cop haguem extret la informació del codi de barres i validat signatures digitals.

Finalment, i amb menys prioritat, també ens interessaria dissenyar un document PDF, que serveixi com a comprovant per al ciutadà un cop ha fet la sol·licitud, i que sigui transmès a l'usuari un cop aquest ha realitzat la sol·licitud.

7. Bibliografia

7.1. *Fonts electròniques*

7.1.1. Gestio documental

[Wikimedia Foundation, Inc](http://es.wikipedia.org/wiki/Gesti%C3%B3n_documental) (28/10/2008). Gestió documental [online]. Disponible en: http://es.wikipedia.org/wiki/Gesti%C3%B3n_documental (2008, novembre)

ECM-SPAIN (30/10/2008). Gestió Documental [online]. Disponible en: http://www.ecm-spain.com/home_sector.asp?Idsector=1009 (2008, noviembre).

Juan Carrión Maroto. Introduccion conceptual a la Gestión del Conocimiento [Online]. Disponible en: <http://www.gestiondelconocimiento.com/introduccion.htm> (2008, Octubre 20)

CANALS, Agustí (2003). "La gestión del conocimiento". En: Acto de presentación libro Gestión del conocimiento (2003: Barcelona) [online]. UOC. Disponible en: <http://www.uoc.edu/dt/20251/index.html> (2008, Octubre 20).

Josep M. Rodríguez Rovira . La gestión del conocimiento: una gran oportunidad [Online]. Disponible en: http://www.elprofesionaldelainformacion.com/contenidos/1999/marzo/la_gestion_del_conocimiento_una_gran_oportunidad.html (2008, Octubre 20).

7.1.2. Documents intel·ligents

Adobe Systems Incorporated (07/08/2008). Adobe Intelligent Document Platform [online]. Disponible en:

<http://www.adobe.com/enterprise/idp.html> (2008, noviembre).

Microsoft Corporation (02/06/2004). Documentos inteligentes en Microsoft Office 2003 [online]. Disponible en: <http://www.microsoft.com/spain/technet/recursos/articulos/1607037.aspx> (2008, noviembre).

7.1.3. Adobe

Adobe Systems Incorporated (2007). Adobe presenta LiveCycle Enterprise Suite [online]. Disponible en: <http://www.adobe.com/es/aboutadobe/pressroom/pr/jun2007/541.pdf> (2008, noviembre).

Adobe Systems Incorporated (07/08/2008). Acrobat Developer Center [online]. Disponible en: <http://www.adobe.com/devnet/acrobat/> (2008, noviembre).

Adobe Systems Incorporated (2006). Foros de usuario a usuario [online]. Disponible en: <http://www.adobeforums.com/webx/.59b6400c> (2008, noviembre).

Adobe Systems Incorporated (2008, octubre 31). Adobe LiveCycle Designer [online]. Disponible en: http://es.wikipedia.org/wiki/Adobe_LiveCycle_Designer (2008, noviembre).

Wikipedia Foundation, Inc (2008, febrer 25), Adobe Flex [online]. Disponible en: http://es.wikipedia.org/wiki/Adobe_Flex (2009, març).

Form Nation (07, Novembre 2008) Executing web service calls in a form guide [online]. Disponible en: <http://blogs.adobe.com/formnation/> (2009, març).

Seidor (marzo, 2009) Soluciones de Gestión integrales y completas para sus procesos de negocio [online]. Disponible a: http://www.seidor.es/irj/portal/anonymous?guest_user=Guest_ES (2009, marzo)

7.1.4. Software gestió documental

[Wikimedia Foundation, Inc](#) (09/10/2008). Software de gestión documental [online]. Disponible en: http://es.wikipedia.org/wiki/Software_de_gesti%C3%B3n_documental (2008, noviembre).

Microsoft Corporation (2008). Información general del producto Microsoft Office InfoPath 2007 [online]. Disponible en: <http://office.microsoft.com/es-es/infopath/HA101656343082.aspx> (2008, noviembre).

openTrends Solucions i Sistemes, S.L. (2008). Gestió de formularis intel·ligents en PDF (openPDF) [online]. Disponible en: http://www.opentrends.net/web/sites/opentrends/CA/Tecnologies/Formularios_PDF/ (2008, noviembre).

[Wikimedia Foundation, Inc](#) (14/10/2008). Scribus[online]. Disponible en: <http://es.wikipedia.org/wiki/Scribus> (2008, noviembre).

[Anduin.net / Øverby Consulting](#). (17/09/2008). Scribus 1.3.3.12 Stable Release Announcement [online]. Disponible en: <http://www.scribus.net/?q=release> (2008, noviembre).

[Wikimedia Foundation, Inc](#) (2008, octubre 23). Microsoft Office Infopath [online]. Disonible en: [http://es.wikipedia.org/wiki/Microsoft_Office_Infopath_\(2008\)](http://es.wikipedia.org/wiki/Microsoft_Office_Infopath_(2008)), noviembre).

Microsoft Corporation (2008). Tipos de archivos gráficos que puede utilizar InfoPath [online]. Disponible en: <http://office.microsoft.com/es-es/infopath/HP010967373082.aspx> (2008, noviembre).

7.1.5. Tutorials Adobe

Adobe Systems Incorporated (2008). Adobe. Centro de recursos de ayuda [online]. Disponible en:
http://help.adobe.com/es_ES/Acrobat/8.0/Professional/help.html?content=WSAB9BF8C6-5263-4bbf-B560-A6DBCB1D0ADB.html (2008, novembre).

Adobe Systems Incorporated (2008). Adobe. Installing and Deploying LiveCycle ES Using Turnkey [online]. Disponible en:
http://help.adobe.com/en_US/livecycle/8.2/installturnkey.pdf (2009, Febrer).

Adobe Systems Incorporated (2008). Adobe. Installing Your Development Environment [online]. Disponible en:
http://help.adobe.com/en_US/livecycle/8.2/install_dev_enviro.pdf (2008, Octubre).

Adobe Systems Incorporated (2008). Adobe. Administering LiveCycle® ES [online]. Disponible en :
http://help.adobe.com/en_US/livecycle/8.2/admin_guide.pdf (2009, Febrer).

Adobe Systems Incorporated (2008). Adobe. Programming with LiveCycle ES [online]. Disponible en:
<http://livedocs.adobe.com/livecycle/8.2/programLC/programmer/help/wwhelp/wwhimpl/js/html/wwhelp.htm?&accessible=true> (2009, Febrer).

Adobe Systems Incorporated (2008). Adobe. LiveCycle® Designer ES Scripting Basics [online]. Disponible en:
http://help.adobe.com/en_US/livecycle/8.2/LiveCycle_Designer_Scripting_Basics.pdf (2008, Juliol).

Adobe Systems Incorporated (2008). Adobe. Learning About LiveCycle Reader Extensions ES [online]. Disponible en:
http://livedocs.adobe.com/livecycle/8.2/eu_readerext/wwhelp/wwhimpl/js/html/wwhelp.htm?&accessible=true (2009, Febrer).

Adobe Systems Incorporated (2008). Adobe. Designer[online]. Disponible en:http://livedocs.adobe.com/livecycle/8.2/acrobat_designer/wwhelp/wwhimpl/common/html/wwhelp.htm?context=Adobe LiveCycle Designer Help&file=000997.html (2009, Febrer).

7.1.6. Altres

Tecno Soft(2008). Código de Barras. Disponible a:
<http://www.tecnosymbol.com/simbologias/PDF417.html> (marzo, 2009).

Monografias.com. Códigos de barras. Disponible a:
<http://www.monografias.com/trabajos42/codigo-de-barras/codigo-de-barras4.shtml> (marzo, 2009).

